LAPPEENRANTA-LAHTI UNIVERSITY OF TECHNOLOGY LUT
School of Engineering Science
Industrial Engineering and Management
Business Analytics

Oskari Lehtonen

ANALYSIS OF PRODUCTION TESTING DATA AND DETECTING ABNORMAL BEHAVIOR

Examiners: Professor Mikael Collan
Post-doctoral Researcher Christoph Lohrmann

TIIVISTELMÄ

Lappeenrannan-Lahden teknillinen yliopisto LUT
School of Engineering Science
Tuotantotalouden koulutusohjelma

Oskari Lehtonen

Tuotetestausdatan analysointi ja poikkeamien havainnointi

Diplomityö
2020
63 sivua, 19 kuvaa, 3 taulukkoa
Tarkastajat: Professori Mikeal Collan ja Tutkijatohtori Christoph Lohrmann

Hakusanat: Koneoppiminen, Ohjaamaton oppiminen, Poikkeamien havaitseminen

Tämä diplomityö esittää metodeja tuotetestauksen kehittämiseen soveltamalla ohjaamattoman oppimisen menetelmiä havaitsemaan poikkeamia tuotteiden testaamisesta kerätystä datasta. Näitä metodeja käytetään case-tutkimuksessa ABB:n tarjoamaan testausdataan heidän Alpha tuotteestaan, joka on teollisuudessa käytetty sähköä käyttävä tuote. Työn tavoitteena on luoda työkalu, jota voidaan käyttää poikkeavien yksilöiden havaitsemiseen yhdessä nykyisen testausprosessin kanssa.

Aikaisemmista tutkimuksista selviää monia lupaavia metodeja ja algoritmeja, joita voidaan hyödyntää poikkeamien tunnistamiseen ohjaamattoman oppimisen menetelmillä. Muuttujien käsittelyyn usein käytetään erilaisia versioita Pääkomponentti analyysistä (PCA) ja Autoenkoodaajista, jotta poikkeavat yksilöt erottuvat selkeämmin normaaleista. Myös näitä metodeja voidaan soveltaa poikkeamien tunnistamiseen mittaamalla, kuinka hyvin ne mallintavat alkuperäistä dataa. Itse poikkeamien tunnistamiseen useimmiten käytetään erilaisia klusterointi-algoritmeja tai yhden luokan luokittimia.

Lopulliseen työkaluun valittiin neljä metodia: Hotelling's T^2 ja Q-residuaali statistiikat, sekä HDBSCAN klusterointi-algoritmi sekä yhden luokan tukivektorikone. Näiden metodien yhteistuloksen perusteella valitaan yksilöt, jotka todetaan poikkeaviksi vähintään kolmella metodilla, poimitaan jatkoanalyysiin. Analyyseihin käytetyistä 1436 yksilöstä 14 todetaan olevan poikkeavia, joka vastaa viallisten tuotteiden odotettua määrää. Näitä yksilöitä tutkimalla voidaan löytää muuttujia, jotka aiheuttavat eroavaisuuksia normaaleihin verrattuna. Tässä työssä tehtyä tutkimusta, metodeja ja kehitettyä työkalua tullaan tulevaisuudessa hyödyntämään ABB:n tuotetestauksen kehittämisessä.

ABSTRACT

Lappeenranta-Lahti University of Technology LUT
School of Engineering Science
Degree Programme in Industrial Engineering and Management

Oskari Lehtonen

Analysis of production testing data and detecting abnormal behavior

Master's thesis
2020
63 pages, 19 figures, 3 tables
Examiners: Professor Mikael Collan and Post-doctoral Researcher Christoph Lohrmann

Keywords: Machine learning, Unsupervised learning, Anomaly detection

This thesis presents methods to improve production testing methods by applying unsupervised machine learning to find anomalies from the data collected during testing. These methods are applied to a real-world case with the company ABB and their product Alpha an industrial electronic product. The goal is to create a tool to detect these deviating samples that can be used together with the current testing process.

The literature review reveals multiple promising methods and algorithms to be used for detecting anomalies in unsupervised manner. For feature extraction purposes different variations of Principal Component Analysis and Autoencoders are used to separate the anomalous samples from the normal. These methods have also been used to detect the anomalous samples by measuring how well they re-construct the data samples. For actual anomaly detection clustering and one-class classifiers are mainly used.

For the actual tool to be created four methods were selected: Hotelling's T^2 statistic, Q residual statistic, a clustering algorithm called HDBSCAN and one-class support vector machine classifier. Results from these methods are combined to determine which samples are determined to be anomalous. It is decided that when three or more methods agree on the sample being anomalous it is taken into further analysis. From the 1436 samples used for the actual analysis 14 samples were deemed anomalous, which corresponds to the expected rate of these products breaking down in the field use. Further analysis of these samples reveals variables that contribute the most to the reason they are deemed abnormal. The research, methods and the tool created in this thesis will in the future be incorporated to improve the production testing process at ABB.

ACKNOWLEDGEMENTS

First, I would like to thank ABB and especially Klaus for his time and effort to initially organize this opportunity for me and Janne for approving this to move forward. We had great plans for the thesis, and everything was exceptionally organized, but then the COVID situation happened. Despite the situation I had all the support needed thanks to Teppo and others involved in this thesis.

Also, a big thank you goes to my family and friends for supporting me through the process. Especially my girlfriend Isa who has been a great joy and support during these challenging times. A huge credit also goes to the great people I got to study with and who inspired me with new ideas.

Lastly of course I want to thank LUT for great education and my instructors for the feedback and guidance regarding to this thesis.

29.11.2020
Oskari Lehtonen

TABLE OF CONTENTS

1 INTRODUCTION

Delivering high quality products has always been one of the key goals that companies pursuit and many standards and frameworks have been created around quality. In the past years new technologies and digital advancements have opened more opportunities to develop quality assurance even further. One of the most trending areas have been machine learning (ML) and artificial intelligence (AI) and how these concepts will revolutionize manufacturing. (Capgemini, 2019) Even though the algorithms and mathematical models have been around for multiple decades, in recent years digitalization has provided multiple platforms to deploy these models and use them to tackle problems in different business areas. The business problem to be solved in this thesis is to help ABB to improve the utilization of the data from the testing process of the product Alpha and to increase the quality observed by the end users of this product. In this thesis models used for detecting samples that differ from the general samples are studied individually and from the perspective of how to gain value in quality assurance in an industrial production environment. During this thesis these samples can be referred for example as anomalous, abnormal, faulty or outliers.

1.1 Purpose of the thesis

This thesis focuses on finding the suitable methods to reduce early field failures by analyzing the data collected from production testing process. Early failures are common with electronical and mechanical products and are usually caused by poor quality of components or mistakes in assembling the products. The purpose is to find abnormal data samples that could indicate an early failure when the product is taken into use. Abnormal samples are detected by implementing different machine learning and analytics methods, used for example in outlier detection. The actual methods used are selected by studying the current literature and research focusing on anomaly detection. As a result of the thesis there is a clear understanding of methods and tools used in outlier detection and also an implementation of a prediction model to the data provided in this case. Previous research is studied in the area of abnormality detection in industrial environment and the algorithms and methods found are also explained in more general context. These studies provide a clear understanding on what methods and tools are most suitable in the context of this thesis and provides the best opportunities to succeed.

The data analyzed in this thesis is created in the testing process of ABBs product Alpha which is an electronic product that is used in industrial applications by the customers of ABB. This type of product is usually used as a part of high-power electrical systems like production machinery, electricity production or powering transportation. The anomaly detection model is developed to this specific product, but the comprehensive literature review gives a solid foundation for the findings and the model to be developed and implemented to also other products.

The value created for the case company by this thesis comes from helping the company to improve the quality observed by the customer by reducing the number of faults occurring in customer use. The quality is improved by detecting more subtle signs in the data from the production testing that nowadays are not detected by using single variable limits. Taking into consideration that the product Alpha can be used in industrial applications, it is clear that if ABB is able to prevent even a few breakdowns beforehand, the cost savings can be considerable in terms of the customers not having to stop their processes and ABB saving in guarantee claims and keeping their customers satisfied. Also, the results of this thesis can be used across the products of the case company and the use cases can be broadened from just product testing to create predictive maintenance applications. These kinds of applications would bring a whole new business case to be sold to the customers of these products.

The problems to be solved in this thesis can be presented as the following research questions:

"What kind of methods have been used in industrial environment to detect abnormal occurrences in processes?"

"Can possible early failures of product Alpha be detected from the current product testing data by using unsupervised analytical methods?"

1.2 Scope and limitations

This thesis is limited to study only the single product Alpha out of many similar products. The data studied is collected from the production testing processes and it contains two years' worth of data. This study focuses on what can be seen from the data without going into details of how the products are manufactured or how they work. In this stage the resulting anomaly detection model must be able to be run on a laptop by a production testing engineer. The model should be able to run in few minutes time when the production testing data is available for the production testing engineer. Due to the current COVID-19 situation everything is done remotely and testing in the actual production site is not possible during this thesis.

1.3 Structure

This thesis is divided into three larger sections: background (Ch. 1-4), literature review (Ch. 5) and the case (Ch. 6-7). The first chapters provide background knowledge for the concepts discussed in the literature review and in the case. The background part goes through basic concepts in quality assurance, outliers and different machine learning methods related to detecting outliers and improving quality assurance. This part describes different types of algorithms in general and gives more detailed explanations for some of the most used methods in anomaly detection.

The literature chapter of the thesis focuses on the research done previously in the area of anomaly detection. The chapter starts with explaining how the relevant research articles were collected to have a sufficient base for the literature review. The literature review itself focuses on different types of unsupervised methods used to detect outliers and abnormal behavior from the data in industrial use cases. Different methods are compared based on the results and their suitability for different types of datasets. From these methods the most suitable are then selected to be used in the case part of the thesis.

The practical part of the thesis describes a case implemented with ABB to detect abnormal units from product testing data. The case part begins with a brief introduction to ABB to provide some context about the environment which the case is implemented in. Some aspects of the

dataset used for the abnormality detection are described and then the methods used are presented. The methods used are selected based on the literature review chapter and are described in a very detailed and mathematical way. Finally, the results of the case are discussed and recommendations for further work and research are given.

2 QUALITY ASSURANCE AND PRODUCTION TESTING

In this chapter some main topics of quality assurance and its development during the years are presented to give background knowledge of why these operations are important and how they are used in practice. The chapter also reflects on how digitalization, new technologies and machine learning can be used to improve production quality.

The terms around managing quality can be divided in multiple ways. The viewpoint used in this thesis is illustrated in the figure 1 below. Quality assurance (QA) is sub-section of quality management and production testing is seen as part of quality control. For the purpose of this thesis only QA and production testing are covered to keep the focus closer to the actual case.

Figure 1 Taxonomy of quality management (ISO, 2015)

Quality assurance can be defined as all the planned and systematic actions implemented in a company that can be shown to increase confidence on fulfilling quality requirements for the customers. (ASQ, 2012) In other words quality assurance aims to make sure that the products manufactured within the company can be confidently be sent the customers without having to worry about large amount of customer returns due broken or faulty products. QA has a long history and different methods for QA are constantly being developed. Around the Second World War first sampling, standardization and statistical methods were used to ensure the quality of military equipment. Since then many frameworks for quality have been developed and one of the most famous is the ISO 9000 series of quality management standards. (ASQ, 2020a) The series provides concepts and principles to help companies implement quality management and assurance systems. Companies following these instructions can be also certified for ISO 9000 series which is acknowledged worldwide. (ISO, 2015)

Based on the definition by ASQ production testing and quality control can be seen as a part of quality assurance. Whereas quality assurance is a broader term, quality control and product testing are more operational techniques. (ASQ, 2012) Nowadays some very common tools for quality control for example are control charts and histograms. Both of these are usually used to track some key measurements, like lead time or some feature in the product, and the idea is to see if some samples differ from the normal. (ASQ, 2020b) Products need to be tested to see that they meet the criteria set by customers and that they are suitable for the tasks they were designed to accomplish. (ASQ, 2012) In this way these actions enable the quality assurance. Especially with products that are used in industrial purposes or in other critical fields, the cost of broken equipment can be very damaging to the company in terms of replacing equipment under warranty or possible lost customers due bad experience.

When customers know that the products, they are looking to buy are high quality it can make the final purchase decision easier and quality is also seen as one of the key elements to increase the value offering for the customer. (Kotler, Armstrong and Opresnik, 2018) This can be achieved by thorough testing and communicating it to the potential customers. One possible way to test the product is to simulate the usage of the product in a controlled environment and see how it reacts. It is also common to simulate conditions and stress, which would not be

expected in normal use, during the test. Some examples could be to use the product in a very hot or cold temperature, overload the recommended capacity or for example run a motor with higher revolutions per minute that its limit would be when installed in a vehicle. According to Lienig and Bruemmer (2017) with this kind of testing, the early failures are attempted to be minimized. It is also stated that inadequate testing correlates with larger number of early failures. Especially for electronic components the number of failures drop down to a fraction after few weeks of continuous operation. (Lienig and Bruemmer, 2017) These kinds of actions are also considered in the testing process of the product Alpha.

Data is usually collected during the testing process from various sources to see if the values that measure the quality or the desired state of the product stay in the limits set by the company. It is also important to keep in mind that not everything that can be measured, needs to be measured. Data can be collected automatically through different types of sensors or they can be measured by hand. Also, visual inspections and other qualitative inspections can be part of the product testing process. Qualitative inspections and other measurements by hand need to have clear instructions and they need always be done in the same way to have reliable results. In addition to the actual quality of the product, these issues also effect to the quality of the data to be analyzed. If the quality of data used for the analysis is bad, the results of the analysis cannot be good either. To tackle some of the issues caused by human error, machine learning solutions can be used to replace simple tasks. Angelopoulos et al. (2019) give multiple examples of machine learning applications in industrial environments. Especially for visual inspections machine vision can be used to detect faults and abnormalities in the product, for example missing pieces or poor paint job. Predictive algorithms can be used to detect if there are some issues in the production process by taking into an account values measured in different parts of the production line. (Angelopoulos *et al.*, 2019)

Overall the advanced methods mentioned above are quite new and still in developing and emergin. In the World Quality Report 2019-20 conducted by Capgemini (2019) the trends included machine learning and artificial intelligence as one of the main trends in quality assurance. Many companies are currently using ML and AI solutions in their quality assurance processes and many are running proof of concept projects to see how these solutions can be utilized. (Capgemini, 2019) Overall the advancements in technologies and digitalization have

produced a fourth industrial revolution focusing, among other things, on cloud computing, internet of things (IoT), machine learning, big data and advanced analytics. (Erboz, 2017) Also When using these technologies in quality control, the amount of data analyzed isn't so restricted anymore in terms of computing and data storing capabilities and more use cases to create business value in the quality processes can be found. Also, with advanced analysis methods the amount information gained from the data can increase by analyzing data in real time and with more complex methods than before.

3 OUTLIERS AND ANOMALIES

Outliers have been defined in multiple ways in the literature, but generally they are defined as observations that differ from the expected pattern or are outside the usual distribution of the measurements. (Bansal, Gaur and Singh, 2016) In the context of this thesis the terms outliers, anomalies and abnormalities are used as synonyms. There are multiple ways to classify observations or groups of observations as outliers depending on the approach or definition used.

Clear outliers can also be detected from visualizations created from the data and usually this is the simplest method to check for outliers when dealing with smaller datasets and with low dimensions. By plotting, it is also possible to see outliers when examining the relation between two variables in addition to one dimensional histogram. In the figure 2 different types of possible outliers and anomalies are shown in plots.

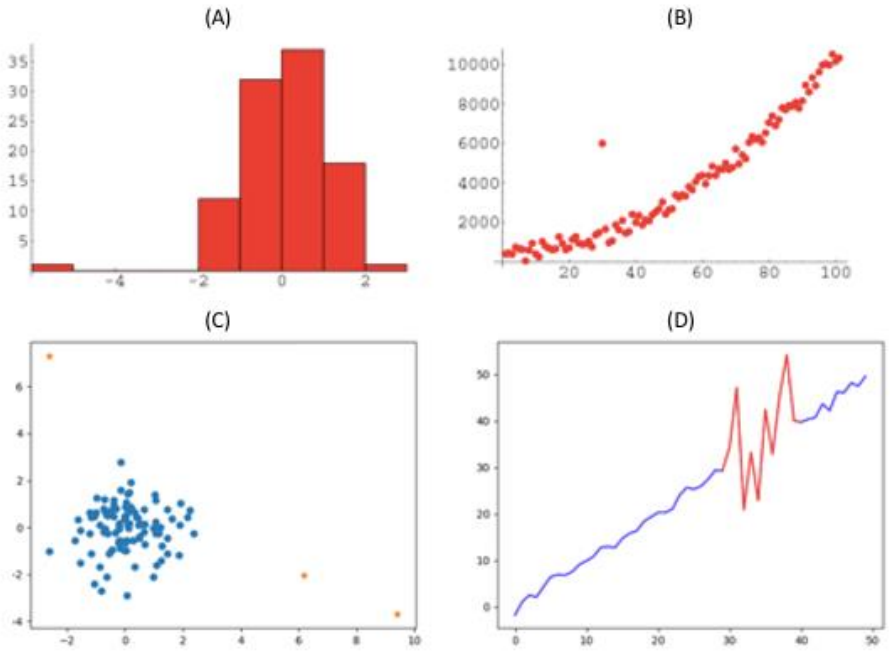

Figure 2 Different types of outliers and anomalies (Renze, 2020; ReNom, 2018)

The plot A in figure 2 represents a distribution of measured values and it shows an outlying sample far from the remaining distribution. This kind of outliers can be detected with the

standard deviation-based method, because it differs significantly from the main population. In the B and C plots, the outliers are caused by unusual value in one of the two variables in relation to the other. In the B plot the values are expected to follow a certain curve, but one datapoint deviates from the curve pattern significantly. Whereas the C plot shows more of a case where both values need to stay in certain limits and for some datapoints the other value is significantly higher and for that reason it is clearly out of the group considered normal. The D plot shows how the pattern abruptly changes, caused by some unexpected occurrence. Here the x-axis represents time. It is important to know what kind of outliers and abnormalities are expected to be found, since different types of outliers need different types of methods to catch them. Also, not all methods necessarily consider same points as outliers.

One commonly used method is to check if the variable is over three standard deviations away from the mean. In practice this means that from the normal distribution, 0.3 percent of the samples are classified as outliers. (Brandon-Jones, Slack and Johnson, 2013) In practice this method is quite limited, since it assumes that the values of variable are normally distributed and can be applied only to single variables at a time. This method is also commonly used in the Six Sigma processes. (Brandon-Jones, Slack and Johnson, 2013)

The outliers presented previously are very simple cases and are in two dimensions at most. When moving to data with larger number of dimensions the methods become more advanced and computationally more demanding. Bansal, Gaur and Singh (2016) describe different types of outliers that can be found in two dimensional and multidimensional data. Two common types are distance and density-based outliers. Distances between two points can be calculated for example using Euclidean distance or Mahalanobis distance in high dimensional datasets. Datapoints that are far away for example from the mean of the points can be classified outlying. In the density-based approach datapoints that are not located in dense regions are classified as outliers. (Bansal, Gaur and Singh, 2016)

There can be multiple causes for the outliers detected. One common cause is a simple error in taking the measurement or when saving it, regardless of if it is taken by hand or through a sensor. In addition to wrong values it is also possible that no data is acquired. These kinds of outliers usually are very clear and can be discarded from the dataset. In the context of this thesis

the distinction between wrongful measurements and possible faulty products needs to be clear. In this thesis the outliers trying to be detected represents an abnormal unit tested in the production line. It is assumed that if a single product differs significantly from the majority there can be something wrong. The testing process already detects single variable cases where the value doesn’t stay between the assigned limits, so more advanced and multivariate methods are needed to detect anomalies.

4 MACHINE LEARNING

In this chapter the basic concepts of machine learning are explained to give a clear understanding on the different types of methods used in anomaly detection and why some methods can be used to detect anomalies, and some cannot. The chapter begins with the concept of preprocessing the data and then moves on to different types of algorithms used in machine learning.

4.1 Data preprocessing

Before applying any algorithms to the data, the data needs to be preprocessed to eliminate the poor performance caused by issues in the data. Preprocessing step includes tasks like cleaning and normalizing the data. Also, what to do with missing values needs to be decided. (García, Herrera and Luengo, 2015) When trying to detect anomalies from data with large number of variables, the feature selection and feature extraction can have considerable effects on separating the anomalous datapoints from the normal ones. But feature selection or extraction doesn't always improve the performance, or the improvement can be very minimal. (Doraisamy *et al.,* 2008) In addition to possible improvements to the ML algorithm performance, other advantages of feature selection and extraction methods are reducing the amount of data and storage needed, making the data easier to visualize and providing a possibility to use simpler models for faster processing. (Kacprzyk *et al.*, 2006)

In feature selection the most relevant features are selected to be kept in the dataset, when the irrelevant and redundant features are discarded. One simple method is to see if some features have zero variance. Also, set of features that correlate completely are redundant to each other and only one of them is necessary to be used in the analysis. (Bolón-Canedo, Alonso-Betanzos and Sánchez-Maroño, 2015) This kind of occurrences can be seen quite rare, since it would mean that variables are exact copies of each other. Other commonly used method is recursively training a classifier and find what variable makes the biggest difference to performance when removed. (Kacprzyk *et al.,* 2006) This kind of method would require class labels for the data samples and would be considered as a supervised method. With feature selection techniques, it is assumed that the features that were relevant in the past, are also relevant in the future.

(Doraisamy *et al.,* 2008) If there is uncertainty about how the variables could behave in the future, feature selection methods can be used later in the future to see if still same features remain relevant.

In feature extraction the goal is to find the most informative set of features that have been created from the original set of features. (Alpaydin, 2010) One of the most used methods for feature extraction is Principal Component Analysis (PCA), which creates a projection of the data that that it would explain the variability in the data as much as possible. (Alpaydin, 2010) As a result, PCA algorithm creates set of new variables, the principal components, and their values, the scores. The principal components are ordered in a way that the first one explains the variability the most between the components, meaning that the component is the best approximation of the data in one dimension. If all components are used, all of the variation of the original data is explained. (Murphy, 2012) Besides the unsupervised methods like PCA, feature extraction methods can also utilize the information provided by the class label. Supervised methods like Linear Discriminant Analysis (LDA) aim to create features that separate the classes as well as possible. (Alpaydin, 2010)

4.2 Machine learning algorithm types

When talking about machine learning models or advanced analytics methods, they can be divided in to at least two different types. The main types are supervised and unsupervised learning methods. (Murphy, 2012) In supervised methods the goal is to learn how the inputs of the algorithm can be mapped to the output. The training data needs to have the inputs and the corresponding labels available to be able to create the model. When training the model, a cost function is used to measure how well the trained model fits to the desired output. (Alpaydin, 2010) Usually outputs are classified into one or more categories, but also regression models, with continuous prediction values, are supervised learning methods. Supervised methods can also be described as predictive models. (Murphy, 2012)

Some common supervised learning methods are support vector machines, different types of neural networks and regression analysis. When detecting abnormalities in the supervised cases the abnormalities are defined beforehand, meaning that there is knowledge about what kind of

outcomes there can be. Example of a supervised application could be a machine vision application using a neural network to detect missing bolts from a picture of a product taken in the production line. The algorithm would have been trained with images that have all the bolts in place and therefore it would detect that some of them are missing.

In unsupervised methods the outputs of the data are unknown, and the goal is to find patterns or groups within the data and gain information from it. Unsupervised learning can also be referred as a descriptive method or as knowledge discovery. (Murphy, 2012) Clustering algorithms are a common example of unsupervised learning and these can be used for example customer segmentation purposes. (Alpaydin, 2010) With unsupervised methods it is much harder to say whether the results were good or bad, since there is no knowledge of the expected output. Also, similar cost functions cannot be used, or accuracy of the classification cannot be measured similar to the supervised learning. Cost function in clustering can measure how well the datapoints are located in the clusters based on the distance between datapoints and cluster centers. (Rebala, Ravi and Churiwala, 2019)

Unsupervised models are very useful in anomaly detection, since there is no need for predetermined classes. There can be cases where there is no knowledge of the possible labels or the labeling of each sample would require considerable amount of manual work. Some common unsupervised methods for anomaly detection are clustering algorithms and variations of PCA based methods. Some research is also done with auto-encoders, for feature extraction and reconstruction. Also, some semi-supervised methods have been tested, where no labeling information is needed, but the data used to train the model needs to represent the normal state of the measurements (Yao *et al.,* 2019).

When deciding what kind of model to use, the problem needs to be clearly defined and the qualities of the dataset needs to be considered. The main point is whether there are the output values available or not. Other considerations are visualized in the figure 3. Also, when going deeper into the model selection it is good to keep in mind the principle known as Occam's razor. The principle states that one should pick the simplest model possible that explains the data at acceptable level (Duda, Hart and Stork, 2012). In practice this means that you shouldn't go with complex models if the problem can be solved with a simpler model.

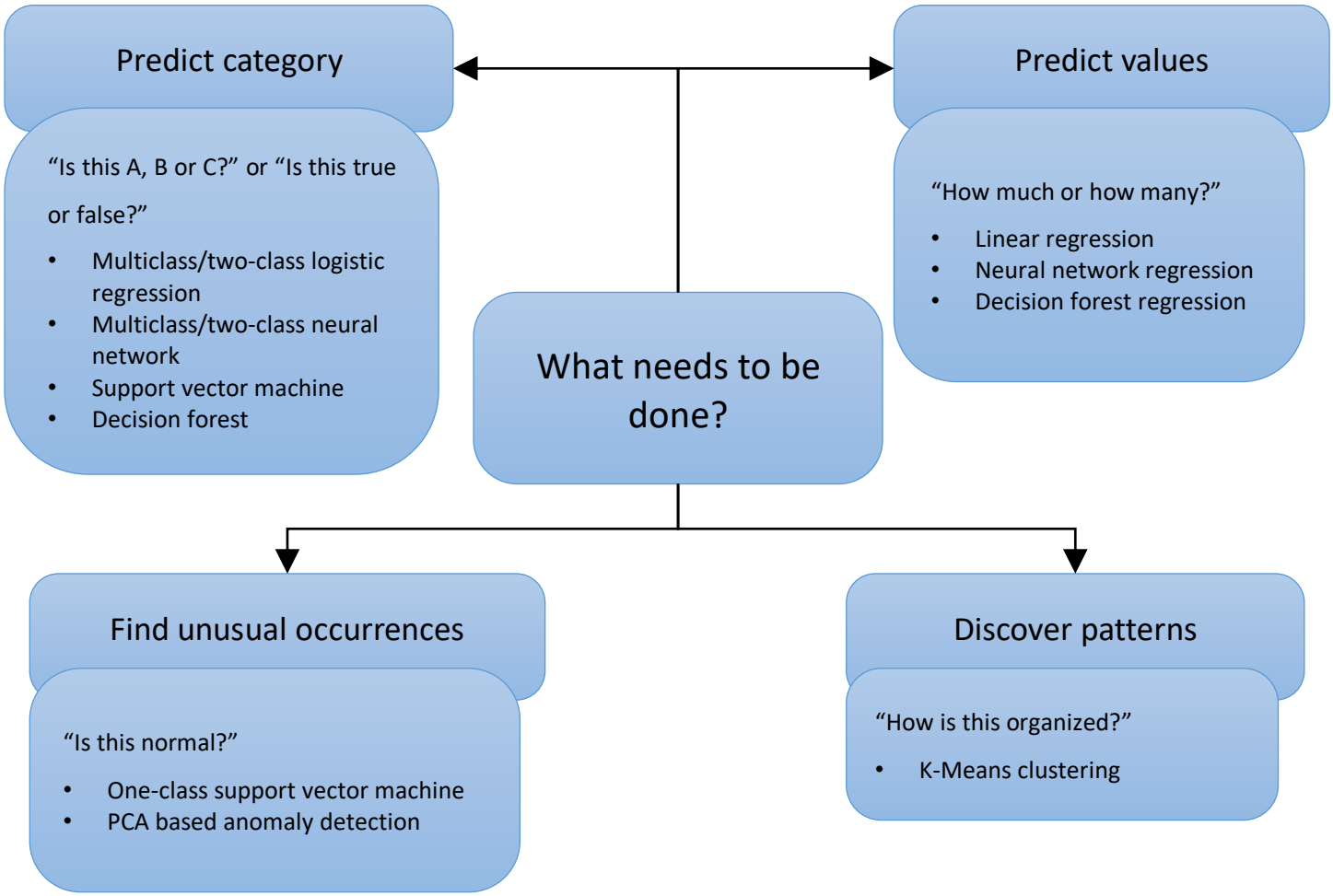

Figure 3 How to choose the best algorithm for the problem in hand. Based on Azure machine learning cheat sheet. (Microsoft, 2019a)

As seen in the figure 3 the starting point needs to be the problem at hand and what needs to be achieved. The figure illustrates a very high-level generalization for the different models and their use cases. Many of the algorithms mentioned can be used to solve many types of problems, but the figure can be seen as a starting point to find models best suited for typical problems trying to be solved by using machine learning.

5 LITERATURE REVIEW

The first part of this chapter describes the information collection process that is done to find relevant articles related to the thesis. Based on the articles selected an overview of researches and projects is created to give an idea of what kind of results have other researchers achieved. The overview is presented in the second part of the chapter.

5.1 State of the art process

The process of identifying relevant literature can be divided into three steps suggested by Webster and Watson (2002).

1. Search is done from databases that include the major journals with the search string defined for the problem in hand. The result of this search can be hundreds of articles.
2. Results are scanned by going through the titles and abstracts. Only relevant articles to the research are included. The result of this can be tens of articles.
3. Last step is to read the main parts of the articles and find important references used in those articles. Those references are then added to the list of articles used.

As a result of that process the relevant articles are found, and those articles should provide a comprehensive base for the research. (Webster and Watson, 2002)

In this thesis the search string used is:

unsupervised (abnormality/anomaly/outlier/fault) detection

The search is done in Scopus database and the result of the search is 2789 articles. From the search results can clearly be seen the current trend in cyber security, since large portion of the articles studied network intrusions and anomalies in network traffic. When trying to find relevant articles for the thesis, the main focus is on industrial and engineering related articles. This is why the search results are then filtered with keywords: “manufacturing”, “production” and “industrial”. After the filtering 328 articles remain to be examined. Next, the type of data

used for the detection is considered when scanning the articles. Researches that solely focused on detecting abnormalities from time series, pictures or videos are mainly discarded. The final articles used as the main sources for the literature review can be seen from the table 1.

Table 1 Articles used as the main sources for literature review

Name	**Algorithm**	**Use case**
High-Accuracy Unsupervised Fault Detection of Industrial Robots Using Current Signal Analysis	Gaussian mixture model K-means	Industrial robots
Outlier Detection in Temporal Spatial Log Data Using Autoencoder for Industry 4.0	Autoencoder	Glass quality inspection
Combining expert knowledge and unsupervised learning techniques for anomaly detection in aircraft flight data	Haar discrete wavelet transform HDBSCAN (hierarchical clustering)	Flight data General
Data-driven anomaly detection using OCSVM with Boundary optimization	CLOF OCSVM	General
Automatic Hyperparameter Tuning Method for Local Outlier Factor, with Applications to Anomaly Detection	LOF OCSVM Isolation forest	General

A Research Study on Unsupervised Machine Learning Algorithms for Early Fault Detection in Predictive Maintenance	PCA K-means Fuzzy C-means HDBSCAN (hierarchical clustering) Gaussian mixture model	Vibration data from exhaust fan
A comparative evaluation of outlier detection algorithms: Experiments and analyses	OCSVM LOF Gaussian mixture model isolation forest Others	General
COMPLEMENTARY SET VARIATIONAL AUTOENCODER FOR SUPERVISED ANOMALY DETECTION	Variational autoencoder	General Air conditioning fault detection
Multiple Component Analysis and Its Application in Process Monitoring with Prior Fault Data	PCA	Process monitoring
Unsupervised anomaly detection based on clustering methods and sensor data on a marine diesel engine	PCA OCSVM HDBSCAN (hierarchical clustering) Gaussian mixture model	Marine engine
Unsupervised Anomaly Detection Using Variational Auto-Encoder based Feature Extraction	KPCA Variational autoencoder	General
Enhancing one-class support vector machines for unsupervised anomaly detection	OCSVM	General

The articles selected to the table below provided either very general possibilities to apply the algorithms to any context or use cases that are closer to an industrial context. These articles provide wide enough scale of different types of algorithms and use cases to find the best methods for the purposes of thesis.

5.2 Unsupervised methods in anomaly detection

The research in unsupervised anomaly detection is mainly divided to methods for feature extraction or selection and methods for the anomaly detection itself. There is a lot of effort put in the research of constructing the features, because it has contributed to better results with the detection of anomalies. The whole process of detecting anomalies is covered in the figure 4 below.

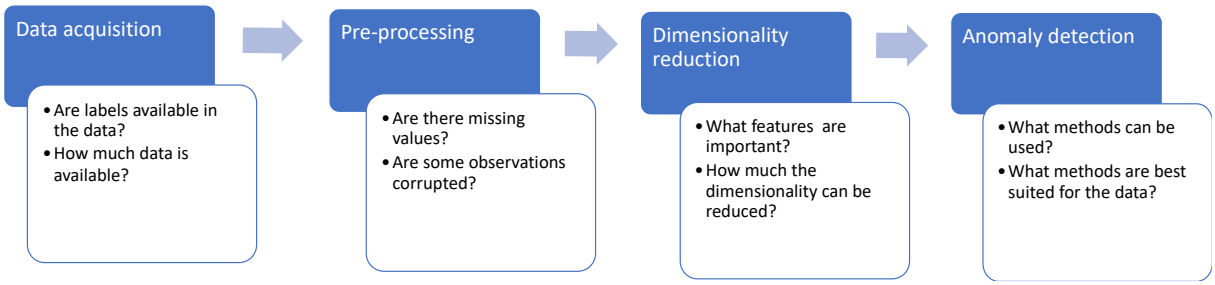

Figure 4 The main steps in anomaly detection process

The data acquisition process or the characteristics of the dataset define whether it is possible to validate the results of unsupervised methods or not. There are multiple possible methods to acquire data in a way that the validation can be done. Cheng et al. (2019) used the equipment in a controlled environment, where they could simulate the abnormal behavior while collecting the data. (Cheng *et al.*, 2019) Biswas (2018) and Kaupp et al. (2019) on the other hand relied on expert knowledge to verify truly anomalous occurrences in their researches. Also many researches focusing on improving algorithms or comparing algorithms on unsupervised anomaly detection methods rely on public datasets like Tennessee Eastman process, which is a realistic industrial process dataset. (Deng and Tian, 2015) Overall, independent on the dataset the methods in these researches remain unsupervised, meaning that the algorithms used have no knowledge of the possible class labels or expected output values.

As previously mentioned, there are multiple ways to choose what variables are chosen to be used in the machine learning process. Vanem and Brandsæter (2019) had over 100 signals where to choose what to use for detecting abnormalities in diesel engines. For initial reduction current engineering knowledge is used to select variables that are relevant to the engine condition. After selecting the relevant variables still dozens of variables remained for their analysis. (Vanem and Brandsæter, 2019) Removing significant number of redundant variables helps to reduce irrelevant noise in the data, but for maximal efficiency more advanced feature selection or dimensionality reduction methods are needed.

Yao et al. (2019) used three different methods for feature extraction: auto-encoder (AE), variational auto-encoder (VAE) and kernel PCA (KPCA). Even though the methods were unsupervised, they had the knowledge of which samples were abnormal. Based on the knowledge they were able to validate the methods. In the figure 5 the results between these three methods can be seen with original 29 variables mapped in to two features. (Yao *et al.*, 2019)

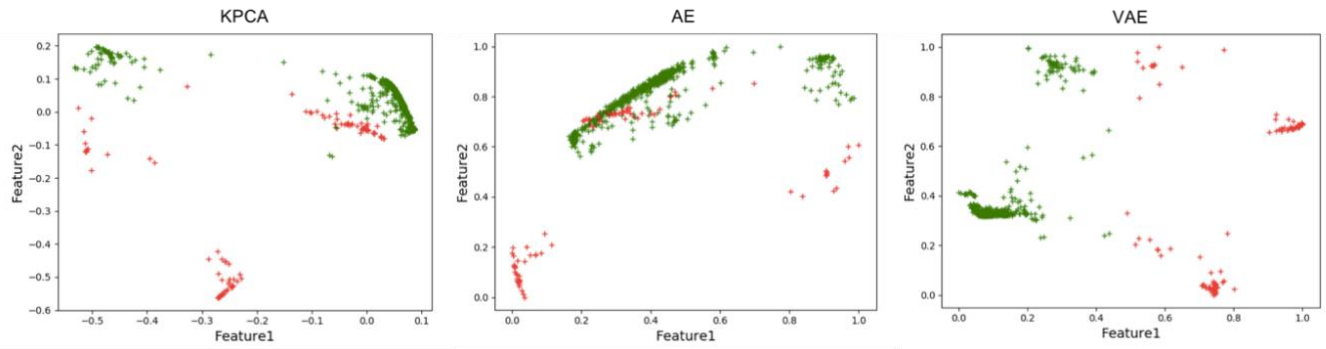

Figure 5 Comparing separation of features between different methods (Yao *et al.*, 2019)

Even though VAE method produces some very good clusters, with truly unsupervised data this kind of results cannot be validated. All of the methods provide clear clusters that could be classified as outliers or abnormalities based on the graphs. Also, the samples between the clearer clusters could be classified as outliers. Although when applying different outlier detection algorithms to these new features constructed, the VAE provided the best results overall with two different datasets. The reason for KPCA to have inferior performance relates to the fact that KPCA discards some of the components deemed unimportant when they really are not.

(Yao *et al.*, 2019) Also, other variations of PCA, dynamic PCA (Russell and Chiang, 2000) and deep PCA (Chen *et al.*, 2018) have been introduced in different studies.

Even though KPCA performed worse than auto-encoders in the research of Yao et al. (2019), there are still reasons for using PCA based methods in feature extraction. Since auto-encoders are a type of neural networks they can be computationally very demanding. Also, PCA can easily be used to reduce the dimensionality of the dataset. In some cases, for example tens of dimensions can be reduced to few PCs with over 95% of the information retained in these PCs (Vanem and Brandsæter, 2019), but with some more complex datasets it takes a lot more PCs to gain enough explained variance. Figure 6 represents two cases where cumulative variance explained behaves very differently. In the plot on the left the first PCs explain the whole dataset much better than in the plot on the right. Both datasets used are example datasets provided by MATLAB.

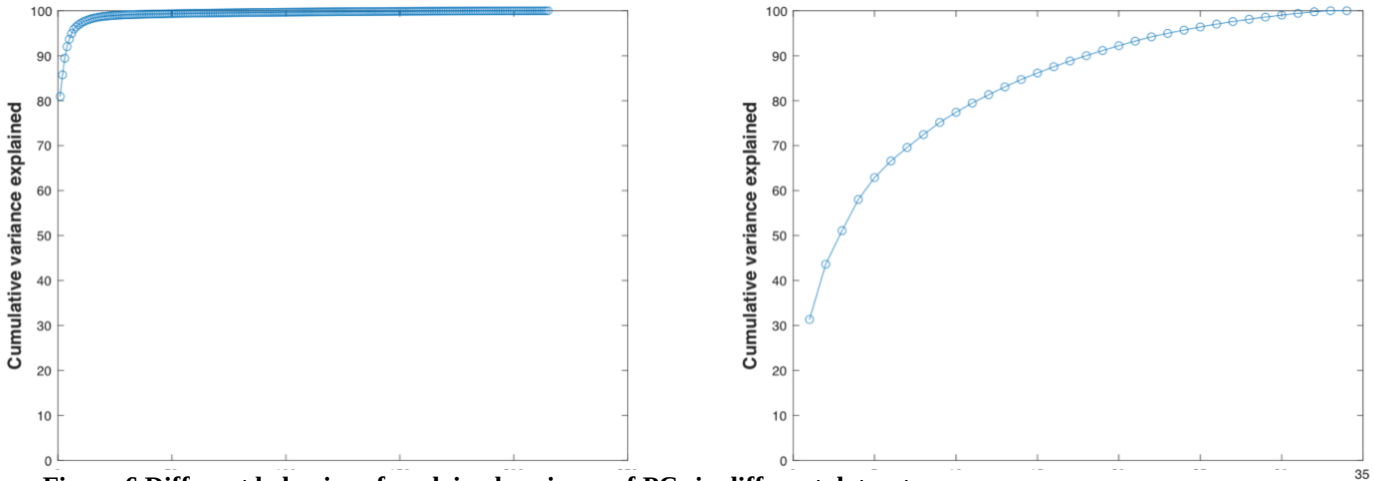

Figure 6 Different behavior of explained variance of PCs in different datasets

While PCA is considered as a dimensionality reduction method it can be used to find outlying datapoints from datasets in a similar manner as with the autoencoders and reconstruction error. Two commonly used methods for detecting outliers are based on Hotelling's T^2 statistic and Q residual values. (Deng and Tian, 2015) Hotelling's T^2 method represents the score outliers in relation to the mean of the scores and it is a multivariate version of the Student's T^2 statistic. The Q residual value on the other hand represents how well the PCA model fits to the datapoint.

In both cases the higher the value the higher the probability for the datapoint to be an outlier. (Wise and Gallagher, 1996) In Figure 7 the use of these values in practice is illustrated in monitoring charts.

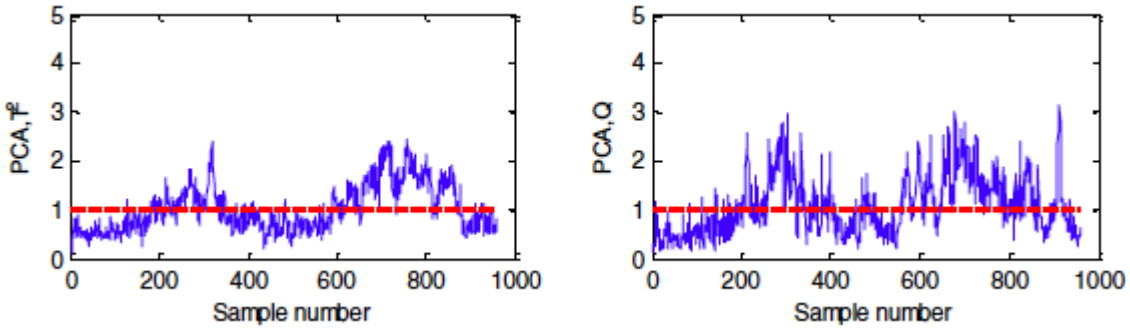

Figure 7 T^2 and Q residual values (Deng and Tian, 2015)

The red lines in Figure 7 represents the confidence limit, which is obtained from a probability distribution. (Deng and Tian, 2015) The values behave similarly, but due the characteristics of the values other samples give higher values in one than in the other metric. It is also possible to plot both values in relation to each other to find the datapoints that are outlying based on both values.

Autoencoders can also be used in anomaly detection similar to PCA without any clustering or classifying algorithms. Kaupp et al. (2019) used autoencoders to detect anomalies by measuring the reconstruction error. Autoencoder is trained with the data collected in the process with the assumption that the number of outliers in the data is very minimal. In practice this means that the trained autoencoder reconstructs the normal samples better and the outlying samples would have a larger reconstruction error. The error is measured by mean squared error (MSE) and the threshold for the error is decided with a domain expert. (Kaupp *et al.*, 2019) Kawachi, Koizumi and Harada (2018) also tested a similar method with VAE. They used the same MNIST dataset as Yao et al. (2019) with similar idea of what is considered to be an anomaly. The results are slightly worse with using only the reconstruction error, compared to the algorithms used by Yao et al. (Kawachi, Koizumi and Harada, 2018) Although these two studies are not completely same so no conclusions on superior method can be stated, since each method accomplishes the expected task well.

In anomaly detection semi-supervised or one class classifiers are widely applied. Support vector machines are usually used for classification tasks for two or more classes but there is also a possibility to use one-class SVMs (OCSVM). In the one-class case the model is trained with the data considered as describing the normal behavior, or at least the number of abnormal samples should be as minimal as possible. (Tax and Duin, 2004) This differs from supervised learning since there is no class labels assigned to the samples, but neither it is fully unsupervised. When the classic SVM tries to find optimal boundaries between the classes, the one-class SVM tries to find optimal boundary where the samples considered normal are inside and abnormalities outside of the border. (Alpaydin, 2010) In practice this would be considered as a two-class case, but the difference comes from the fact that the abnormal samples don't need to be similar with each other. Practical example of two class case could be classifying animals into cats and dogs, when with the OCSVM the classifier would only say whether the animal is a dog or not.

When using one-class SVMs in their research Amer, Goldstein and Abdennadher (2013) noticed significant sensitivity to outliers in the model, meaning that if the training set has significant number of outliers the normal samples are not correctly detected. To make the model more suitable for unsupervised learning they implemented two different methods: Robust one-class SVMs and eta one-class SVMs.

The changes in these new versions are quite small, but they change the outcome considerably. In robust one-class SVMs the idea is to change the goal from minimizing a variable called slack variable to assigning it a value based on the distance from the center of normal samples. The new idea reduces the effect of the outliers, but in theory there can be a case where all of the data points are labeled as outliers. With eta one-class SVMs the slack variables are still minimized in the objective function, but a new variable is introduced to control the contribution of the slack variable. In practice the new variable represents the normality of the data point. The variable is optimized in the process, and ideally the value for outlying samples would be zero. (Amer, Goldstein and Abdennadher, 2013)

In the previously mentioned research, the modified SVMs were compared against normal one-class SVMs and nine other algorithms with four different datasets. Compared to other algorithms all of the SVMs performed better overall and the eta one-class versions was the best. In two of the datasets used SVMs performance was notably better. The accuracy of SVMs varied between 99.8% and 98.3%, except in one data set where all of the algorithms tested had accuracy of 90% or below. After the SVMs a standard k-nearest neighbor clustering algorithm gave the best results overall. Similar results with standard one-class SVMs was also observed in the research by Yao et al. (2019) when comparing it to other algorithms: KNN slightly overperforms the standard OCSVM, but KNN needs to have knowledge of the data labels. The most notable improvement between the modified SVMs and the standard is in time efficiency since they need much smaller number of support vectors. (Amer, Goldstein and Abdennadher, 2013)

A survey done by Alam et al. (2020) shows that research on modifying OCSVM algorithms is not uncommon in the area of anomaly detection. The survey describes over ten different types of OCSVMs that have in some way achieved better results compared to the standard version. Mostly these changes focus on minimizing the effects of outliers in the training data or implementing softer boundaries to the classification where the samples can belong to both classes to some extent. The survey also covers the estimation of parameters for the algorithm, feature selection and how to pick samples for the training process (Alam *et al.,* 2020). The survey shows that OCSVMs can be used in a variety of applications and possibilities for further development are vast. From the survey no single type of algorithm for anomaly detection and feature selection can be selected, due the high dependency on the application and the characteristics of the data.

Since OCSVM is a semi-supervised method, requires certain type of data and gives only one-class results, clustering algorithms provide more freedom when considering the data and use case at hand. Clustering can be used in anomaly detection, since their goal is to group similar data points together. In anomaly detection this would mean grouping normal samples into one cluster and outliers to one or more clusters. There are multiple algorithms for clustering, and they are based on different measurement of the similarity. Similarity of a datapoint can be determined for example by their distance from another or based on how densely the data points

are located in the feature space. In addition to clustering data points to find different groups, also features can be clustered to find similarities. (Murphy, 2012)

Mack et al (2018) used hierarchical clustering in order to find abnormal occurrences from flight operation data. The method is general in nature and can be implemented in industrial applications also. (Mack *et al.*, 2018) In hierarchical clustering there are two ways to start the process. Either the clustering is started with one cluster and then starting to divide it to smaller ones, or the other option is to start with each sample being its own cluster and then combining them into bigger clusters. (Alpaydin, 2010) When clustering the flight operation data, it is assumed that the abnormal samples form a much smaller cluster than the normal samples. In this case the first two clusters formed divided the samples in a way that the other cluster had only 2.5% of the samples in it. The smaller cluster was confirmed by domain experts that it indeed represents the abnormal cases. By using hierarchical clustering, the cluster deemed abnormal can be further divided into smaller clusters and different types of anomalies can be identified. (Mack *et al.*, 2018) Amruthnath and Gupta also used hierarchical clustering in a similar way on a preventive maintenance application. The clustering resulted in three main clusters that represent healthy, warning and faulty samples. (Amruthnath and Gupta, 2018)

Hierarchical clustering method can be visualized in dendograms. Graphical illustration of the results of Mack et al. (2018) is shown in figure 8 below. Here the solid black line represents where the division into clusters is done and each cluster is represented with different color. At the bottom each line ending represents a one sample. When the number of samples is very high and number of possible clusters rise, this kind of visualizations can become very cluttered.

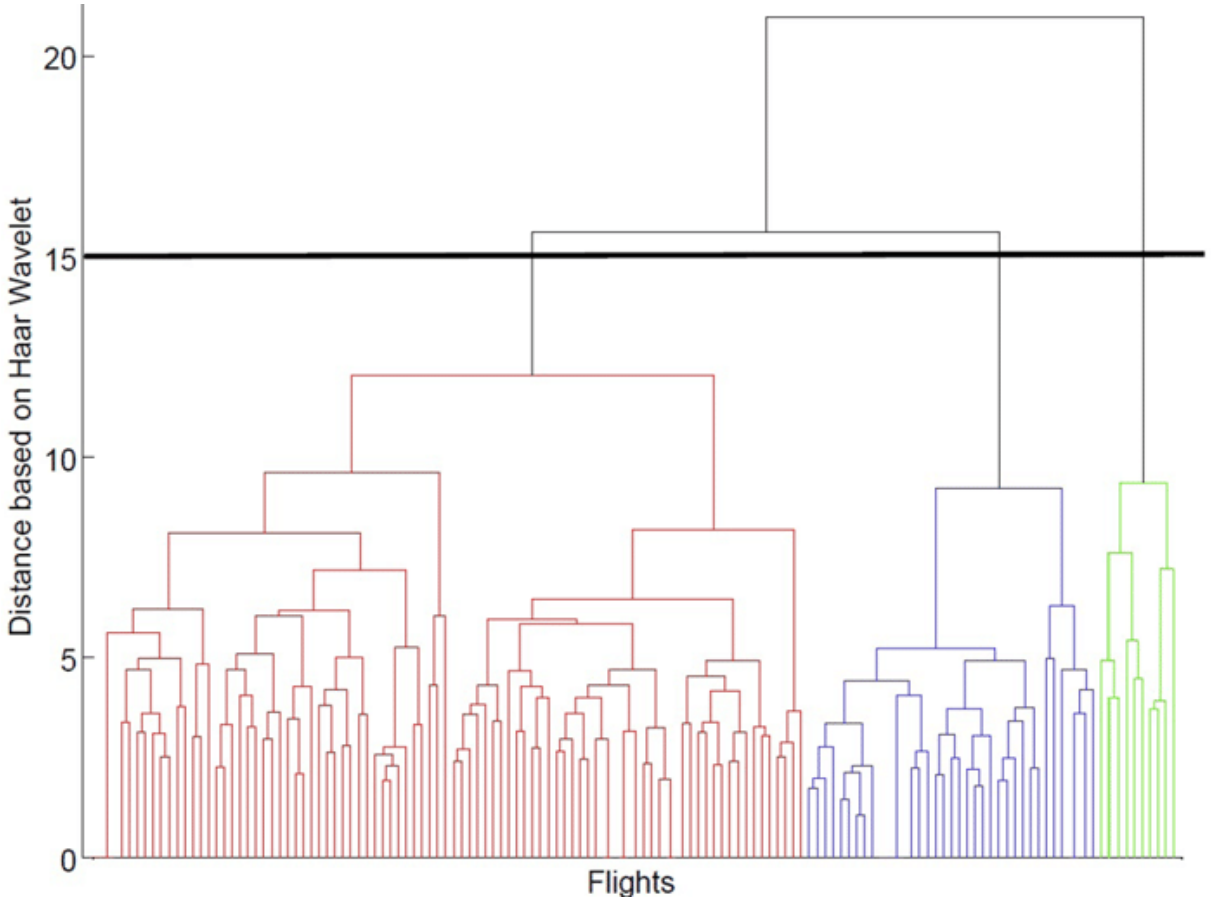

Figure 8 Dendogram of a clustering result of the flight data (Mack *et al.*, 2018)

Other clustering algorithm used in anomaly detection is the K-means algorithm. K-means algorithm needs to have the number of clusters defined and then it iteratively aims to find the optimal clusters. (Rebala, Ravi and Churiwala, 2019) The need for defining the number of clusters can raise an issue, even though when detecting anomalies, the two classes would be abnormal and normal. The issue is caused by the possibility that there are different types of anomalies or even differences in the normal samples. In some cases, the anomalies could be more similar to the normal samples than the other anomalous samples. Also, K-means algorithm assumes that the clusters are convex in shape. (Scikit-learn, 2020) To tackle these issues there are methods to estimate the optimal number of clusters.

If the responsibility of the decision on how many clusters can be found is left for the algorithm, Gaussian Mixture Models (GMM) clustering can be considered. This method does not require the number of clusters defined and it is based on probability distributions. Due to the

probabilistic nature, the clustering is usually conducted as a soft clustering, where the data point can locate in multiple clusters and have different probabilities assigned for belonging to each cluster. (Murphy, 2012) When Cheng et al. (2019) used GMM for anomaly detection it performed better than the K-means algorithm but, for Amruthnath and Gupta (2018) the results didn't notably differ between GMM, K-means and hierarchical clustering. On the latter research the T^2 statistic detected the anomalies better than the clustering algorithms, but with clustering more information about the anomaly can be obtained. (Amruthnath and Gupta, 2018) The figure 9 below represents how different algorithms behave on different shapes of clusters.

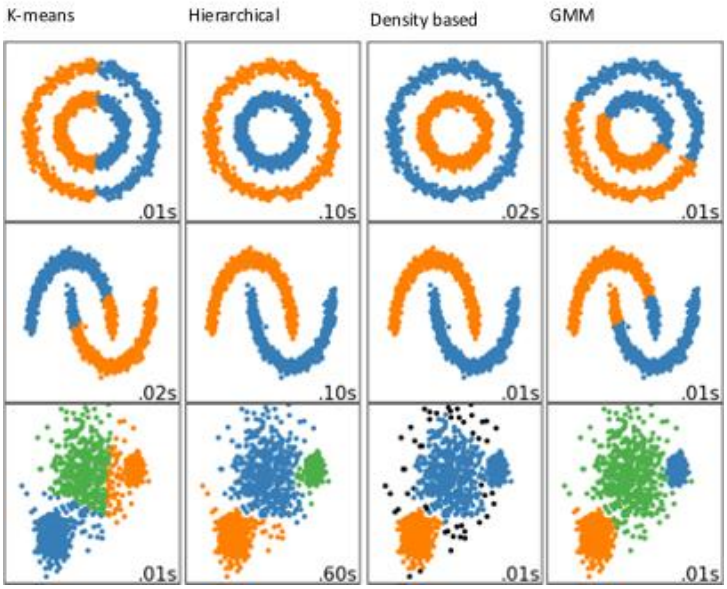

Figure 9 Comparison of clustering algorithms (Scikit-learn, 2020)

The figure 9 illustrates how K-means and GMM behave with non-convex clusters. Hierarchical and density-based methods take the different shapes of clusters in consideration much better, because of the nature on how they link the samples to each other. In this case the density-based algorithm used is DBSCAN. Even though the algorithm doesn't find three clusters in the last case, there are parameters that can be adjusted to create more smaller clusters. (Scikit-learn, 2020) DBSCAN also classifies some samples as noise/outliers. Vanem and Brandsæter (2019) used this quality of the algorithms in anomaly detection. They noticed that adjusting the parameters of the algotihms, the number of clusters and number of detected anomalies changed,

but most of the anomalies detected were the same regardless of the parameters. (Vanem and Brandsæter, 2019)

The variety of clustering algorithms in general have also generated algorithms particularly for outlier detection. One method proposed is called Local Outlier Factor (LOF) which is somewhat related to density-based clustering. The benefit of this method is that it gives a degree of the samples being an outlier and not just a clustering result. Also, being developed for outlier detection it is optimized to detect outliers where clustering algorithms try to find the optimal clusters. (Breunig *et al.*, 2000) Even though LOF is optimized for outlier detection, based on a research by Domingues et al. (2018), many other algorithms including OCSVM and GMM performed better in outlier detection than LOF. The experiments were conducted on 15 different datasets and run multiple times. (Domingues *et al.,* 2018)

Many methods need parameters defined for the algorithms, like the number of clusters or number of points to form a cluster. As a part of their research on anomaly detection, Vanem and Brandsæter (2019) studied how the results vary when different parameter values are assigned and presented some methods for selecting the right values. With each algorithm by changing the possible parameters the sensitivity for anomalies also changed, but mainly the same samples were selected as anomalies with all parameters. When using methods that use different parameters the results need to be validated with domain experts to have a clear view of what parameters work the best. (Vanem and Brandsæter, 2019)

Since the methods are unsupervised there is no way to know which anomalies detected are false alarms. To tackle this problem few algorithms can be run parallel to see which datapoints are detected by all of the algorithms. Also, an expert opinion on what percentage of the observations could be anomalous can be used as a reference to validate the algorithm performance. Expert opinion can also be used for selecting the right number of clusters, based on the assumed number of different types of anomalies. (Vanem and Brandsæter, 2019)

With unsupervised anomaly detection feature extraction and selection is a key part of the process, since the better the separation between an anomaly and a normal datapoint, the easier it is for the detection also. Although, with unsupervised methods the goodness of the separation

cannot really be seen without some validation, but for example one can measure on how well the two classes are separeted. Overall in the unsupervised anomaly detection research, some key methods are PCA based methods, clustering methods and one-class classifiers. No clear superior algorithm can be decided due the different behavior in different kind of data. For this reason, also a set of algorithms should be used when trying to detect anomalies in truly unsupervised manner. Also, it was seen that with adequate knowledge in the algorithms and the data, by fine tuning the algorithms some performance improvements are achieved.

6 CASE: DETECTING ABNORMAL BEHAVIOR FROM PRODUCT TESTING DATA

In this part of the thesis the information from the literature review section is used in a real-life case with the company ABB. The case starts with understanding the business case and having a clear view of the problem that needs to be solved. The next steps are to introduce the data to be analyzed and perform exploratory data analysis to gain information from it. The last steps of this case are to implement different analytical methods for anomaly detection and to analyze the results of these methods.

6.1 Introduction to ABB

ABB is described as a multinational company that focuses on driving the digital transformation of industries. ABB focuses on four business areas: Electrification, Industrial Automation, Motion and Robotics & Discrete Automation. ABB operates in over 100 countries and has around 147 thousand employees. The current ABB brand was created in 1988 when two companies with over 100 years of experience merged into one company. Currently ABB has its headquarters in Zurich, Switzerland and their stocks are listed in multiple exchanges. (ABB, 2020b)

In Finland ABB has operations in around 20 cities and factories in Helsinki, Vaasa, Hamina and Porvoo. Product categories represented in Finland include for example motors, generators and drives. With about 5400 employees in various positions, ABB is one of the biggest industrial employers in Finland. (ABB, 2020a)

6.2 Defining the problem

The problem to be solved arises from the need to find more subtle signals in the testing data to find possible problems within the different units of product Alpha. Products that pass the extensive testing process can still break down in the early stages of the use at customer sites. Since the current testing process has limits for each value measured, the assumption is that more

advanced and multivariate methods can be used to find more subtle deviations that imply possible faults in the product.

The usage of the tool for analyzing the testing data would be to run the analysis after testing process has finished and to see if alarms come up. If something abnormal is detected a deeper analysis of the reasons and possible changes can be done. Then the product can be tested again and see if similar issues arise again. Because the current testing process is quite good, the tool must not generate too many false alarms.

The solving of the problem has been limited to unsupervised methods because of the lack of the data in the outcomes of the products when put into field use. Information from the units broken down at the customer is available for some time periods but it is not enough to train models.

6.3 Introduction to the dataset

The dataset available for the purposes of this thesis is acquired from ABB production testing database of product Alpha. The data in total has over 9000 rows and it has been collected from the year 2018 to early 2020. The total number of variables is 299 and the dataset has a unique ID column. For the first 5000 samples collected during 2018 the information, if the specific unit has broken down in early stages of field use, is available in the dataset. In this case early field failures happen approximately in the first year of use. That is why this kind of information is not available for the latest samples. These units can possibly be used to validate the efficiency of the selected methods in terms of how accurately they detect the faulty samples which breakdown in the early stages of field use. The percentage of faulty units in the 5000 samples is 1% which can also be used as a benchmark value for the methods. There have also been some changes in the testing process during the timeframe where the data has been collected, so for the building of the models and initial analysis only the latest 1442 samples are used which are collected in 2020. Then the results and models based on those units are applied to the rest of the data is possible. All of the analysis is done in a Jupyter Notebook using Python programming language.

The dataset includes all of the data collected during the testing process of the product Alpha, so it includes multiple different types of variables. Most of the variables are continuous numerical values describing attributes like time in seconds, temperatures and electric currents. Some variables describe possible errors in the process in a form of fault codes. Usually this kind of codes are long sequences of integers which can affect the analysis when considered as numerical values. With the help of domain experts, the redundant variables that are known to be irrelevant are removed from the dataset. This reduces the number of variables to 265 from the initial 299 to ease the actual data analysis.

First the dataset is analyzed for missing values. A quick analysis of the number of missing values in different variables reveals that some of the variables have over 50% of the values missing. Some of the missing values can be explained with the introduction of new variables over time to the testing process, since when examining the latest data, the percentage of missing values decreases noticeably. For this dataset a clear threshold for removing these variables can be found. The number of missing values decreases from thousands of samples to 360 samples per variable quite steadily, but then the number drops to six. All of the variables having more than or equal to 360 missing values are then removed. In this case 360 missing values represent 4% of the total number of samples. This reduces the number of variables by 56 to 209. Similar analysis is also done for individual samples. This reveals that the previously noticed variables with six missing values are six samples that are corrupted in some way and are missing most of their values. Initially if there would have been some individual values missing for some unknown reasons, the variables would have been removed by deciding a percentage of what amount of missing values are tolerated within variables and then imputed with column means.

When the variance of the variables is analyzed it can be noted that some of the variables have zero variance. These variables are also removed from the dataset and now the total number of variables remaining is 179. It can also be noted that nine variables have a considerably higher variance compared to the rest of the variables due the larger scale of possible values. This is also why normalization needs to be used to bring the variance to the same scale. All of these variables represent measurements of duration. To have a better idea of the variation of the data some of the variables with the highest variation are plotted in histograms. These histograms can be seen in the figure 10 below.

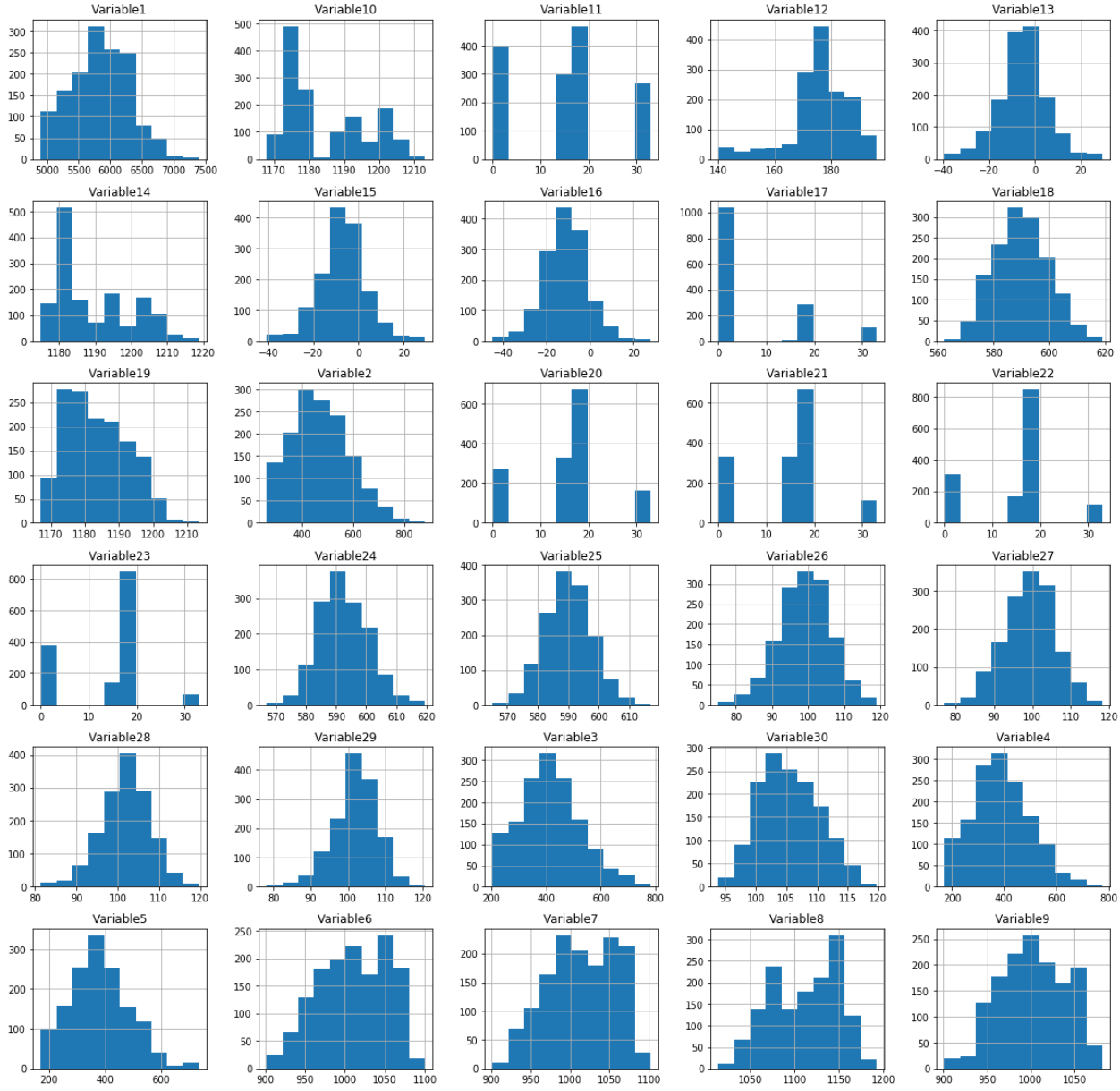

Figure 10 Histograms of the 30 most variating variables

The last step before applying the actual analysis methods is to normalize the data. The data is normalized for each variable to have a zero mean and standard deviation of one. This method is called the z-score normalization. (García, Herrera and Luengo, 2015) On addition to being general practice with machine learning, it is also needed for the PCA transformation to yield better results, because the method is based on maximizing the variance. For the actual analysis after these preprocessing steps the dataset has 1436 rows and 179 columns.

6.4 Methods

In this section the methods used for detecting the anomalies are explained in detail before applying them to the dataset in hand. The first method introduced is the Principal Component Analysis used for preprocessing and it also works as a basis for Hotelling's T^2 and Q-residual statistics. The next methods explained are the one-class support vector machines and HDBSCAN clustering used for detecting the abnormal samples. The purposes and use cases of these methods are introduced in the literature review chapter and this chapter focuses on explaining the actual algorithms and mathematics behind them.

6.4.1 Principal Component Analysis

Explaining Principal Component Analysis can be started by considering **X** as the data matrix with I rows and J columns, where rows are considered as samples and columns as variables. Single samples can be denoted as x_i (i = 1,…,I) and columns as x_j (j = 1,…,J). Since the idea of PCA is to map the data to new coordinate system as a linear combination, the variables can be written as (Bro and Smilde, 2014)

$$t = w_1 \times x_1 + \cdots + w_J \times x_J \quad (1)$$

and a matrix notation of this would then be

$$t = \boldsymbol{X}\boldsymbol{w} \quad (2)$$

Here $\boldsymbol{w}$ is a vector with $\boldsymbol{w}_j$ elements considered as weights or coefficients and **t** is a new vector, also referred as scores, in the same space as the x variables retaining most variation possible from the original data. The goal is to maximize the variance of vector t by choosing the optimal weights ($\boldsymbol{w}_1 \ldots \boldsymbol{w}_j$). To prevent selecting arbitrary large numbers for $\boldsymbol{w}$ it is constrained to be a unit vector. (Bro and Smilde, 2014) Therefore, the problem for the first component can be written as

$$\underset{\|w\|=1}{\operatorname{argmax}}(\boldsymbol{t}^{\mathrm{T}}\boldsymbol{t}) = \underset{\|w\|=1}{\operatorname{argmax}}(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{w}) \tag{3}$$

and the optimal $\boldsymbol{w}$ can be found as the first eigenvector of $\mathbf{X}^{\mathbf{T}}\mathbf{X}$. The rest of the principal components can be found by subtracting the already calculated components from **X** and repeating the process. (Bro and Smilde, 2014)

One way to estimate how well the **t** replaces **X** is to calculate the explained variation. This can be done by calculating the sum of variance of all the principal components and then calculating the ratio between the total variance and the variance of the component at hand. The variation explained can be used to choose the number of Principal Components used for the rest of the analysis. (Bro and Smilde, 2014)

Next the statistics used for detecting abnormalities in the data are explained. First the Hotelling's T^2 is calculated from the scores ($\boldsymbol{t}$) and it can be seen as an extension of the t-test. (Bro and Smilde, 2014) The actual calculation is

$$T_i^2 = \frac{\boldsymbol{t}_i^T(\boldsymbol{T}^{\mathrm{T}}\boldsymbol{T})^{-\mathbf{1}}\boldsymbol{t}_i}{I-1} \tag{4}$$

Where T is the matrix of scores from PCA and $\boldsymbol{t}_i$ is a vector of the scores of the *i*th sample. If it is assumed that the scores are normally distributed, then it is also possible to calculate confidence limits for the values. The other statistic used with PCA is the Q-statistic, which is calculated as the sum of squared residuals of the samples. (Bro and Smilde, 2014) These statistics are used to measure how well the PCA transformation fits the data and therefore it can be used to detect samples that fit poorly to the model compared to the majority of the data.

6.4.2 One-class Support Vector Machine

The one-class SVM is a modification from the classical SVM. SVMs are mainly used for problems where there are two classes that are linearly separable. The model tries to create a hyperplane between the classes, with as wide of a margin as possible. SVMs can still be used for regression problems and linearly inseparable problems with slight modifications and special properties. (Rebala, Ravi and Churiwala, 2019)

When classical SVMs aims to create an optimal boundary between classes, the one-class version tries to maximize the boundary from the origin. In this case the samples between the boundary and the origin are considered as normal and other samples as abnormal. Also, OCSVM doesn't need class labels for training the boundary. (Guo *et al.*, 2018)

In mathematical notation the OCSVM optimization problem can be written as a quadratic program, that need to be solved, seen below. (Schölkopf *et al.*, 2001)

$$\min_{w \in F, \xi_i \in \mathbb{R}^n, \rho \in \mathbb{R}} \left(\frac{1}{2} \|w\|^2 + \frac{1}{\nu n} \sum_{i=1}^{n} \xi_i - \rho \right) \quad (5)$$

$$\text{subject to } \left(w \cdot \phi(x_i)\right) \geq \rho - \xi_i \text{ and } \xi_i \geq 0 \; \forall i$$

Here *w* is normal vector of the hyperplane and ρ is the intercept of the hyperplane. ξ_i is a relaxation factor that makes the distances from support vectors to optimal classification boundary adaptable. OCSVM also needs a parameter to estimate the percent of abnormal samples which is represented by *v* and it can have values from greater than 0 to 1. (Guo *et al.*, 2018) The value *v* is later referenced as the nu parameter when this algorithm is used in Python.

The problem can be described as a quadratic programming problem which can be solved by using the Lagrange function.

$$L_{\mathrm{p}} = \frac{1}{2}\|\mathrm{w}\|^2 + \frac{1}{vn}\sum_{i=1}^{n}\xi_i\beta_i - \sum_{i=1}^{n}\left(\left(\mathrm{w}\cdot\phi\left(\mathrm{x}_i\right)\right) - \rho + \xi_i\right)\alpha_i \qquad (6)$$

and with minimizing variables w, ρ and ξ_i by setting their derivatives to zero obtains formula which provides the final decision function:

$$f\left(\mathrm{x}\right) = \mathrm{sgn}\left(\sum_{i=1}^{n}\alpha_i K\left(\mathrm{x}_i,\mathrm{x}\right) - \rho\right) \qquad (7)$$

here the $K(x_i,x)$ represents the kernel function used to map the features in to another feature space. (Guo *et al.,* 2018) The sgn represents the sign function, which means that the result can only be one or minus one in this case. The result is one when the value is positive or zero and minus one when the value is negative. Here the minus one represents the abnormal samples. (Schölkopf *et al.,* 2001)

6.4.3 HDBSCAN

HDBSCAN is a hierarchical clustering algorithm developed by Campello, Moulavi, and Sander (2013) and it extends the DBSCAN density-based clustering algorithm by enabling the hierarchical clustering. Where density-based clustering tries to find areas where datapoints are densely populated, the hierarchical method uses similarity measures like generally used Euclidean distance. (Alpaydin, 2010) The HDBSCAN algorithm also has the ability to recognize outlying samples as noise and leave them out from the clusters detected. (Campello, Ricardo J. G. B., Moulavi and Sander, 2013)

First the algorithm estimates the densities in the data by transforming the space. For this purpose, the core distance is defined, which represents the distance to the *k*th nearest neighbor of point *x* and denoted as $core_k(x)$. The distance metric used generally is the Euclidean distance, but other metrics can also be used. The larger the distance is the lower the density of the data is

at point *x*. To spread apart those low density points a distance metric called the mutual reachability distance is defined:

$$d_{mreach-k}(a,b) = max\{core_k(a), core_k(b), d(a,b)\} \quad (8)$$

here the *d(a,b)* is the original distance between *a* and *b*. This allows the distance of dense points to remain the same, but the sparser points are pushed away to their core distances. This helps to separate the sparse samples from the expected clusters. The mutual reachability distance is calculated between all points resulting in a $n \times n$ pairwise matrix. (Campello, Ricardo J. G. B., Moulavi and Sander, 2013)

Next step of the algorithm is to build an extended Minimum Spanning Tree (MST), which is a undirected graph where the edges between nodes are weighted. (Campello, Ricardo J. G. B. *et al.*, 2015) In this case the datapoints are connected by the mutual reachability distances and the extension adds a connection to the datapoint itself as the core distance value. From here the hierarchical clustering can be started by removing edges between points in descending order. In this way first all of the points are in a single cluster and when going the hierarchy levels the end result is each datapoint being its' own cluster. (Campello, Ricardo J. G. B., Moulavi and Sander, 2013)

The outlier detection ability of the algorithm is based on the clustering result and each datapoint is assigned a value to represent the degree of how likely the datapoint is an outlier. The datapoints are compared to the densest datapoint of the density-based hierarchy in the closest cluster to determine the outlier score. The formula for calculating the outlier score for sample x_i can be seen below. (Campello, Ricardo J. G. B. *et al.*, 2015)

$$\frac{\frac{1}{d_{core}(x_l)} - \frac{1}{d_{core}(x_i)}}{\frac{1}{d_{core}(x_l)}} \quad (9)$$

Here the sample x_l represents the densest sample mentioned earlier. If the sample x_i is in the dense area of the cluster the value results in a very small number. And in the other hand when the sample is in the sparser areas, the core distance is larger and therefore the outlier score will be higher. (Campello, Ricardo J. G. B. *et al.*, 2015)

6.5 Detecting abnormalities

The actual process of detecting anomalies begins with applying the PCA to the cleaned and normalized data and analyzing the PCs. The first thing after the transformation is to see how much of the variance is explained by the components. From the PCs the first 40 explain 90% of the variation and the first 55 variables 95% of the variation in the data. The accumulation of the variance in proportion to the PCs can be seen in the figure 11.

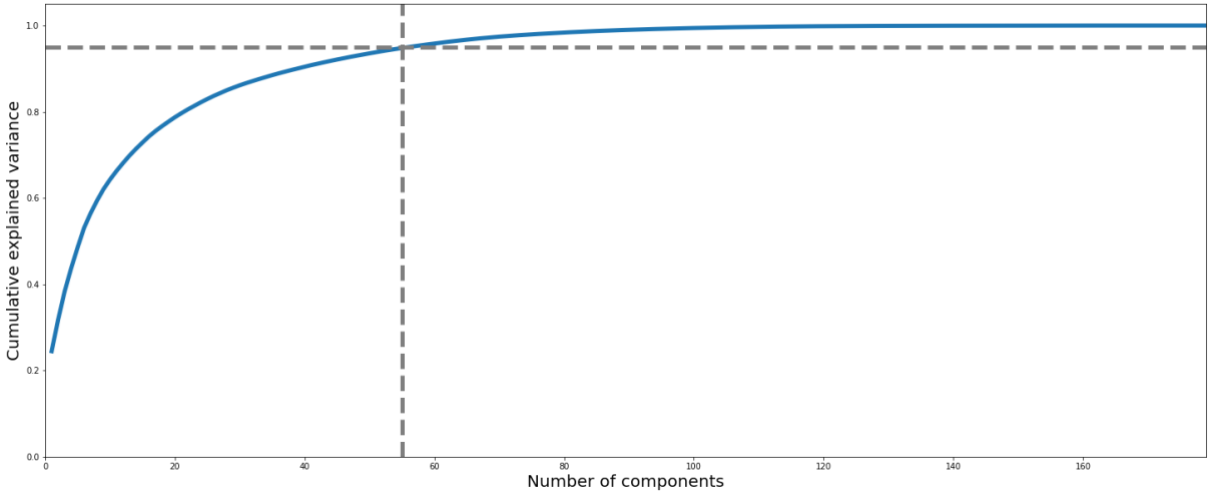

Figure 11 Cumulative variance explained by the principal components

As mentioned in the literature review part of the thesis, PCA is not only used for preprocessing but its characteristics can also be used for anomaly detection. The first statistic used is the Hotelling's T^2 statistic. A value has been calculated for each of the samples and they are visualized in the figure 12. The dashed line represents the 95th percentile of the sample values. That percentile also determines that the samples with higher value than this are deemed as the abnormal samples, since the higher value means that the sample differs more from the rest. The 95th percentile is selected for this and the Q residual statistic instead of 99th because not exactly one percent is wanted because it is only an estimate. Also 95th allows wider margin for not to

miss possible anomalies when the results of all methods used are examined collectively. A total of 72 samples from the 1436 are deemed abnormal by this method. The plot shows some variation, but no major shifts or trends are visible. Most of the samples vary between 120 to 220, which can be seen as normal variation due different conditions or differences in sensors. On top of the clear spikes in the data, at the 1200 units mark a small cluster of units exceeding the threshold value can be seen. This could indicate that there have been some issues with the manufacturing process or in the testing process of the units.

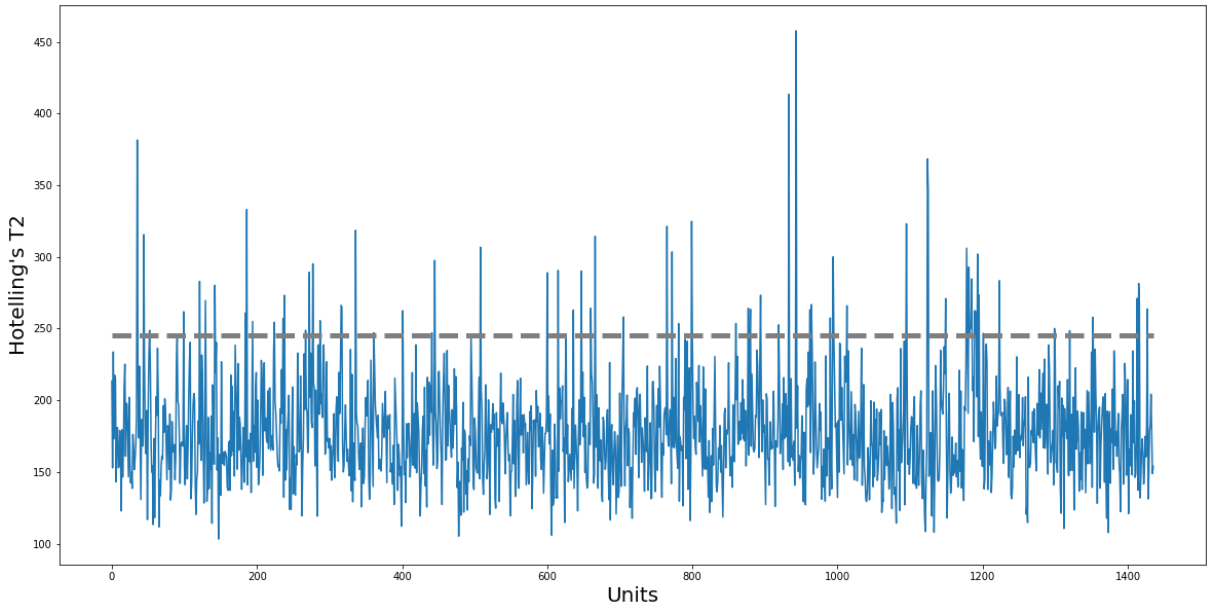

Figure 12 Hotelling's T^2 values plotted for the samples

The other similar statistic used is the Q residual. The values are plotted in figure 13 in a similar way as the T^2 statistic. Also, similar patterns can be seen here: variation exists, but no clear trends. Also, the values are very small because the residual is calculated from all of the PCs. If less PCs are used to calculate the residuals, the values would be higher, but it doesn't affect the general idea of the method. With this statistic also the 95^{th} percentile limit is used and therefore also 72 abnormalities are found. However, when comparing them to the units detected by T^2 there is very little resemblance. Only eight of the units are detected by both of the statistics and for example no similar cluster can be found around the 1200 units mark.

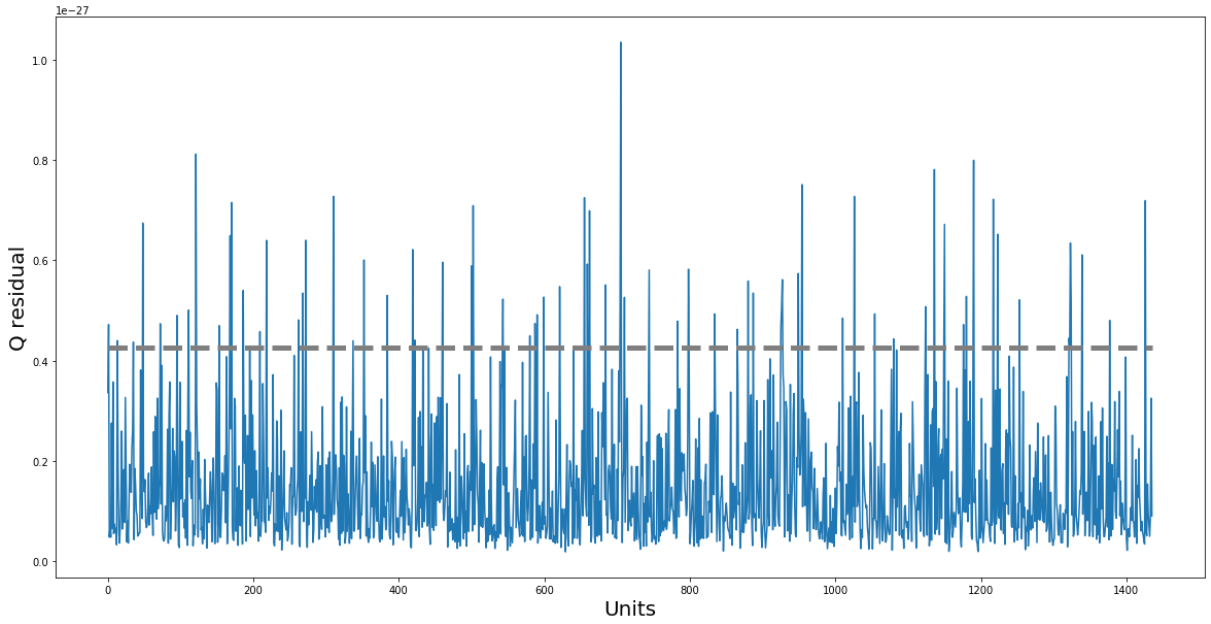

Figure 13 Q residual values plotted for the samples

The values from both of the methods are scaled and plotted in relation to each other in figure 14. The samples deemed abnormal by both measures are highlighted in orange dots. Also, the distribution of the samples resembles normal distribution, but for both measures “a long tail” can be seen. This tail is caused by the abnormal values exceeding the thresholds. As the expected number of faulty products is somewhere around 1% the individual statistics were set to find 5%, which can be seen as clearly too high. Then the combined result of eight samples is 0,6% which is much closer to that estimate.

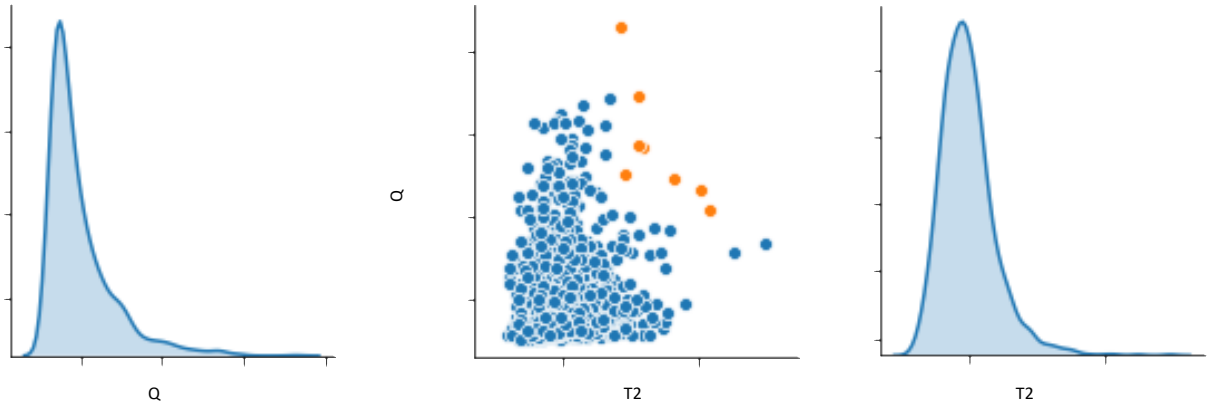

Figure 14 Q and T2 distributions and their relationship

The first machine learning algorithm to be applied for the data is the OCSVM. The data used for the algorithm are the scores obtained from the PCA transformation. The number of PCs used in the OCSVM are limited to 55, describing 95 percent of the variation. Only 5% of the information is lost, but the number of variables is reduced to less than one third of the original. This reduces the number of data used in the calculations of the algorithm significantly and therefore improving the calculation efficiency.

All of the 1436 samples are used to train the algorithm. As the method is unsupervised, there is no way to validate the results with this dataset. The algorithm has input parameters that can be altered to fine tune the results. In this case the most important parameter is the “nu” parameter, which is used to approximate the amount of outliers in the data. The parameter is set to 0.01 based on the number of known faults which were available for the first part of the whole dataset. The kernel used is radial basis function and gamma parameter is calculated by 1/(number of variables * variance). An SVM is generally a fast model to train, but with this amount of data the training happens instantly.

After the model is trained it can be used to predict the label for samples. The model assigns -1 to the samples deemed outlying and 1 for normal samples. The total number of samples deemed abnormal are 56, which means that 3.9% of the samples are detected as abnormal. This differs from the expected 1% by over 41 units. Changing the parameter “nu” to a smaller number doesn’t change the results considerably, so the estimated 0.01 is decided to be the final value for the parameter nu.

The other algorithm used is the hierarchical and density-based clustering algorithm, HDBSCAN. The same training set is used for this algorithm as were before. The algorithm assigns samples either to clusters or labels them as outliers. The outliers are also assigned a score that describes how likely the sample is to be an outlier. The initial assumption is that samples labeled as outliers are the possible abnormal units. There also can be a possibility that the abnormal units create a cluster of their own and this needs to be studied once the clustering is done.

HDBSCAN also has parameters that need to be set before the clustering can be done. The primary parameters to select are minimum size of the clusters and minimum samples which effects on how conservative the clustering is. When experimenting with minimum cluster size with the values over 10 the algorithm assigns all of the datapoints to one cluster. Because of this the minimum samples parameter needs to be set to 1 for the algorithm to pick more subtle clusters and outliers that would have been merged into other clusters when larger values are used.

The clustering algorithm results in three clusters which includes the outlier "cluster". The division to clusters is concluded by the algorithm. 25 samples are detected as outliers and only 16 samples are assigned to one of the other clusters. Therefore, rest of the 1395 samples form a significantly larger cluster. Based on the expected value of faulty units, both the outliers and the smaller cluster could represent those faulty units. This is something that needs to be taken into consideration when examining the results from all of the methods, because the other methods only decide whether the sample is an outlier or not by determining if the sample differs from the rest. HDBSCAN on the other hand can assign those value to an own cluster and detect some other samples as outliers.

6.6 Results of the case

Earlier it was mentioned that part of the data had known outcomes on whether the units have failed in the field use or not. When these methods, which were trained with the most recent 1400 samples, are used for the whole 9000 data samples it becomes very clear that the testing process has changed during the time the data has been collected. Same variables are used, and the same preprocessing steps are taken. The number of missing values in different columns and in individual samples rise considerably. Where the whole data has over 9000 samples, the number of removed samples due multiple missing values is almost 3000. Also, the T2 statistic shows clear shifts in the data in multiple timeframes and approximately the latest 1500 samples follow a very similar pattern within each other. This amount is also very close to the number of samples used to train the models and analyze the process. These changes in the testing

process also make redundant the known faulty samples that are available with the first 5000 samples and they cannot be used to validate the performance of the methods used.

Each method yielded different types of results and classified a different amount samples as abnormal. HDSBSCAN detected the least amount of abnormalities with 25 samples and the OCSVM second least with 56. The 16 samples assigned to one cluster by HDBSCAN did not have any common samples deemed abnormal by the other methods, so it can be stated that those samples would represent normal units. The most sensitive methods to detect samples as outlier are the Q and T^2 statistics based on the PCA transformation. Those are also the only methods where there are no actual parameters for the algorithms. The only thing that can be altered with those methods is the percentile selected. For example, by selecting to use 99^{th} percentile the number of abnormal samples drops down to 15, but the 95^{th} percentile is used in this case because it gives more possibilities for potential anomalies.

The results of each method are combined to a single table for easy comparison between methods. In the table for each sample that has been detected as abnormal a value of -1 is assigned. Different methods assign different values to the normal samples, so they need to be replaced by zeros. This gives the option to sum the values by rows and see which samples have the smallest row totals. In this case the minimum is -4 meaning that all four methods agree that the sample is abnormal. The distribution of samples that have been detected as abnormal at least by one method can be seen in the figure 15.

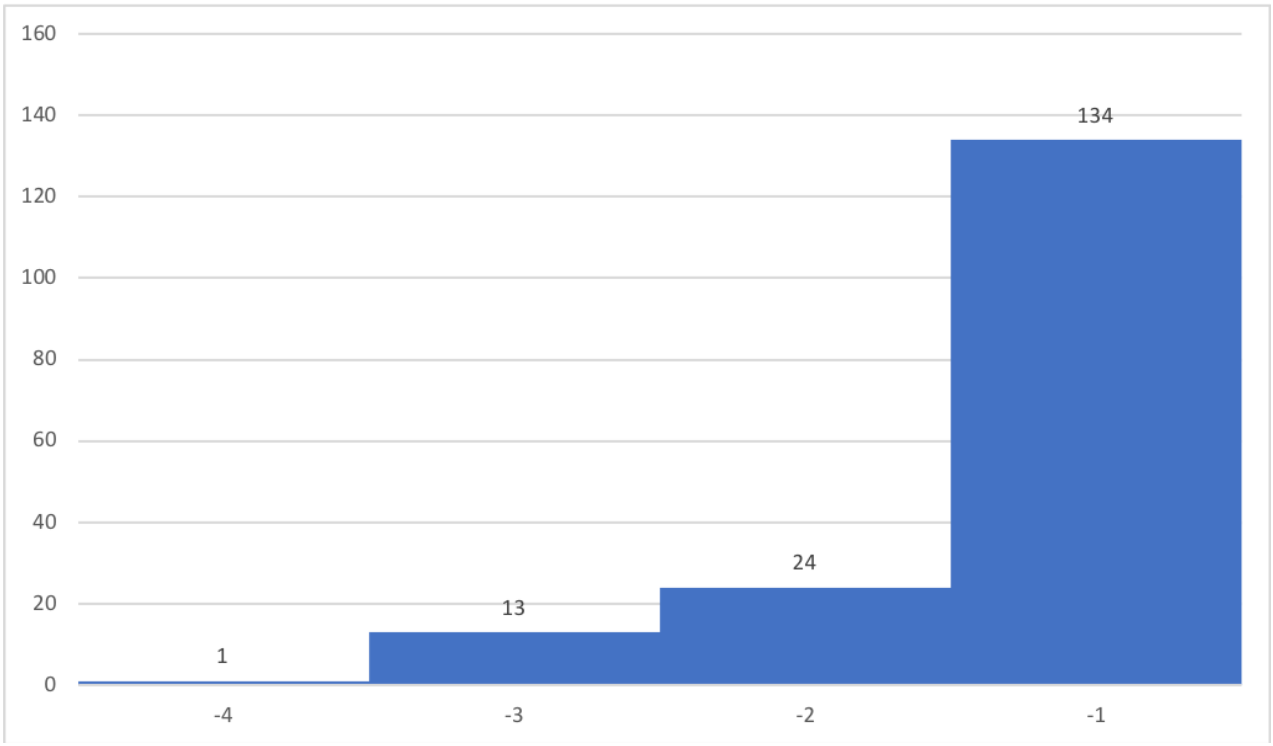

Figure 15 Distribution of samples alerted by atleast one method

A total of 172 samples are detected as abnormal, but only one sample has been detected by all of the four methods. On the other end 134 samples are detected by at least one method, but when considering the -2 group the decrease is very notable. Also, if all samples detected would be considered, it would cover almost 12% of the total samples. This differs from the expected percentage of faulty units so drastically that it can be stated that using only one method is not sufficient to detect actual abnormalities and is sensitive to the noise in the data. When leaving the -1 group out of consideration the cumulative percentage drops to 2,65% which is much closer to the expected value. All of the cumulative percentages and totals can be seen in the table 2 below for each of the groups.

Table 2 Amounts and percentages of detected anomalies

Label	**-4**	**-3**	**-2**	**-1**
Number of samples	1	13	24	134
Cumulative number of samples	1	14	38	172
Percentage of total	0,07 %	0,91 %	1,67 %	9,33 %
Cumulative percentage of total	0,07 %	0,97 %	2,65 %	11,98 %

To see if there are some chronological patterns the samples deemed abnormal are plotted in the order of the testing. The figure 16 below shows that the samples deemed abnormal are spread evenly throughout the reference period. Length of each line represents how many methods have deemed the sample abnormal and the colors represent individual methods. It can be seen that around the 1200 units mark there is some sort of concentration. This pattern is mainly caused by the Hotelling's T^2 statistics which was seen also in the figure 12.

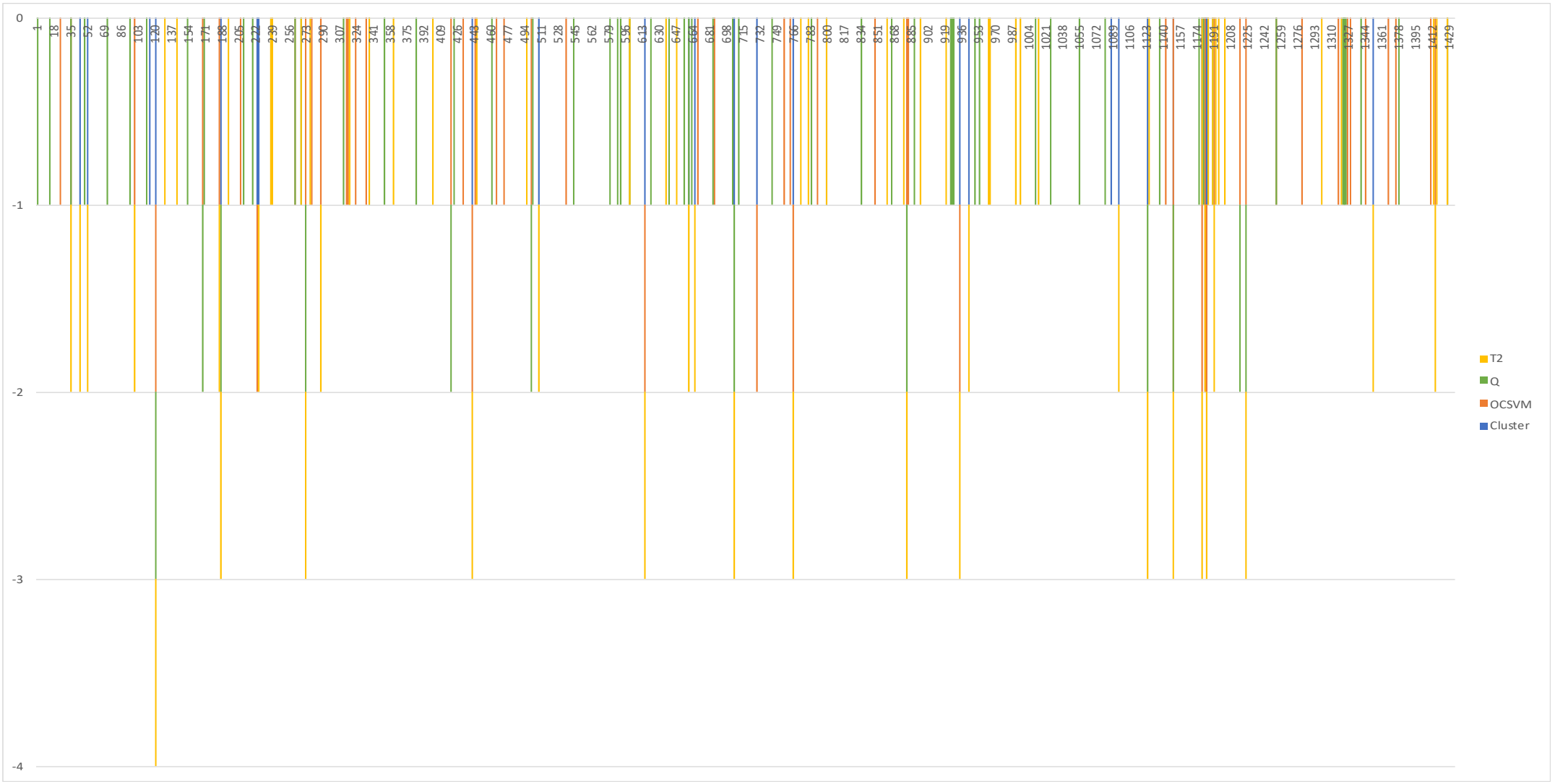

Figure 16 Chronological representation of samples deemed abnormal

When considering the expected value of possible faulty units, the groups -4 and -3 are the most interesting and worth further analysis. First of all, these samples combined are very close to the expected fault rate and also the fact that at least three different types of analytical methods agree on that there is something abnormal in these samples seems to provide reliable differentiation between samples normal and abnormal samples. From the 13 samples in group -3, the Q residual method didn't alert from seven samples where HDBSCAN and OCSVM both did not agree on only three samples to be abnormal. More detailed information about how the results of the methods divide between the groups -4 and -3 is shown in the table 3 below. The fields marked with "x" represents that the sample has been detected abnormal by the corresponding method.

Table 3 Division of the 14 samples deemed abnormal between different mehods

SerialNumber	HDBSCAN result	OCSVM result	T2 result	Q result
1	**x**	**x**	**x**	**x**
2	**x**	**0**	**x**	**x**
3	**x**	**0**	**x**	**x**
4	**x**	**0**	**x**	**x**
5	**x**	**x**	**x**	**0**
6	**x**	**x**	**x**	**0**
7	**x**	**x**	**x**	**0**
8	**x**	**x**	**x**	**0**
9	**x**	**x**	**x**	**0**
10	**x**	**x**	**x**	**0**
11	**x**	**x**	**x**	**0**
12	**0**	**x**	**x**	**x**
13	**0**	**x**	**x**	**x**
14	**0**	**x**	**x**	**x**

Samples in these groups need to be analyzed to see what has caused them to differ from rest of the samples. Analyzing the causes provides an understanding of why the samples have been deemed abnormal and if some actions can be taken to reduce this kind samples. The samples

are divided into normal and abnormal samples by deciding that samples which are deemed abnormal by four or three of the methods are abnormal. In addition to the samples that were not deemed abnormal by any of the methods, the samples alerted only by one or two methods, are all considered as normal. To have some kind of an idea how the abnormal samples differ from the rest, the first six PCs are plotted in relation to each other and the abnormal samples are highlighted in figure 16 below. These six principal components explain 53% of the variation in the data.

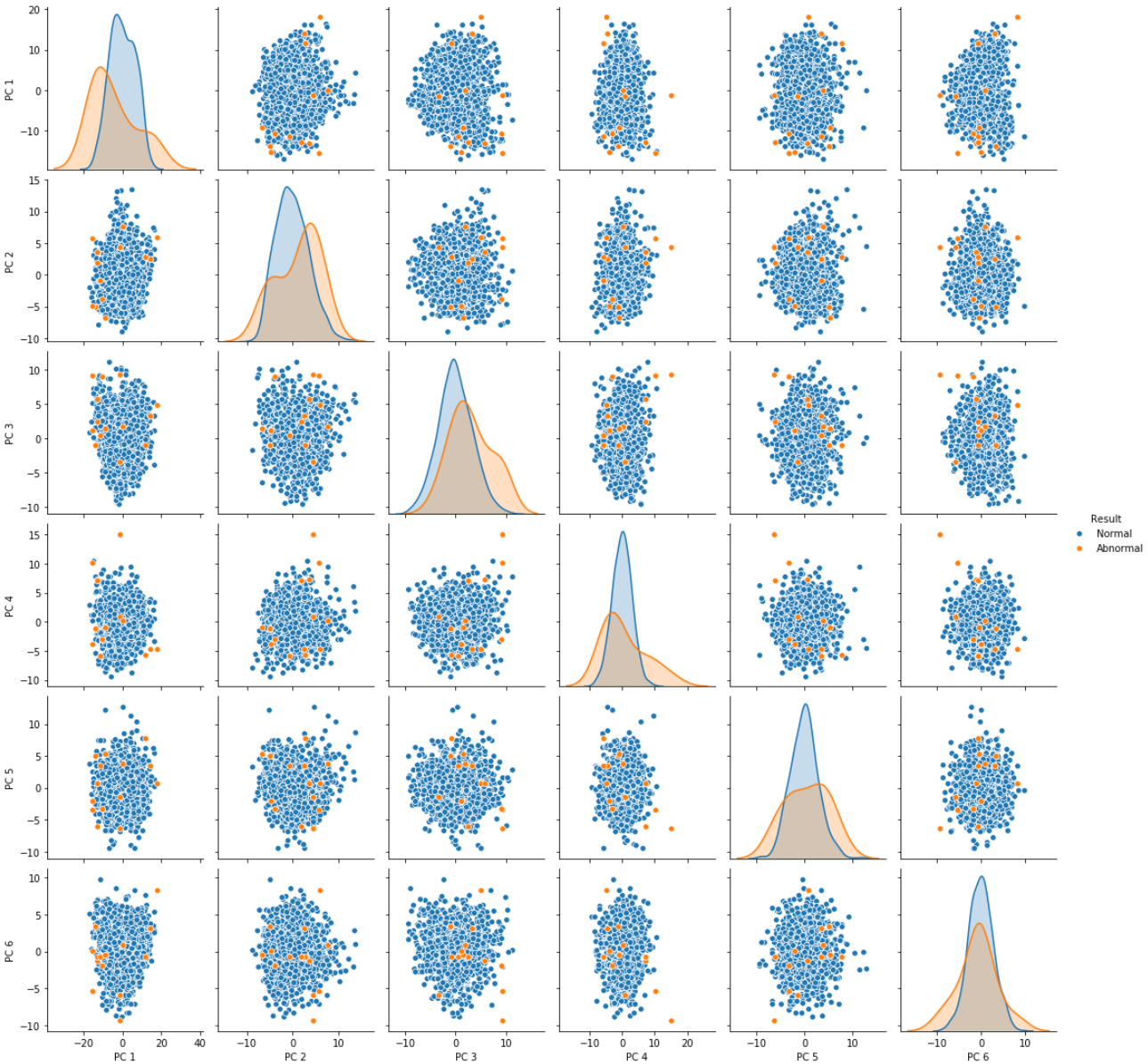

Figure 17 First six principal components with abnormal samples highlighted

When looking at the figure 17, it can be seen that the samples deemed abnormal don't particularly stand out in the individual graphs. In the graphs there can be seen normal samples further from the main group than the abnormal samples and no clear patterns or clusters of abnormal samples can be seen across all PCs. Still with some PCs some clusters and clearly separate samples can be found. For example, with the fourth PC one sample is clearly outlying from the other samples. The differences between the groups can be seen more clearly from the probability distribution plot shown diagonally in the figure. Almost with all of the six PCs the distribution with abnormal samples is spread wider than the with the normal samples. However, all of the distributions are considerably overlapping, and the peaks of the distributions are not distinctly separated.

To get a better understanding what causes theses samples to be detected as abnormal the original variables are analyzed between normal and abnormal groups. First the means of each variables between the two groups are calculated and compared. The comparison is done by analyzing which variables have the biggest difference in the averages between groups. This can indicate possible causes of why the samples have been deemed abnormal. The distributions of 12 variables with the largest difference are plotted in the figure 18 below.

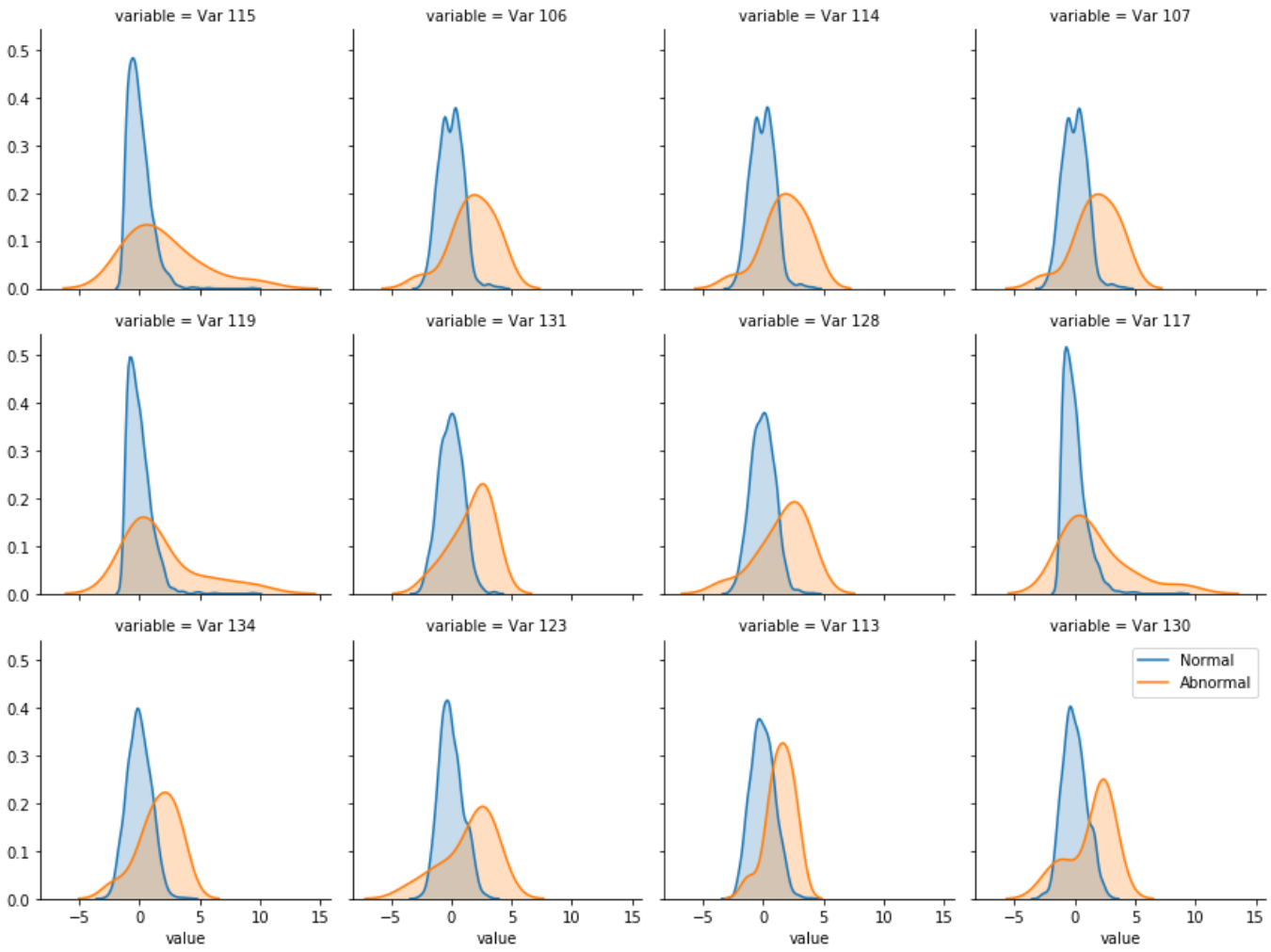

Figure 18 Distributions of samples deemed normal and abnormal

The ideal situation to look in the graphs would be that the two distributions would overlap as little as possible and then also the peaks would be clearly separated. In this case unfortunately the distribution of abnormal samples completely overlaps the range of normal samples, meaning that all of the values that normal samples have, could also be considered abnormal. However, when looking at the distributions of abnormal samples it can be seen that they are much wider spread than the normal samples. This means that there are at least some values that can be only found in abnormal samples. Still it is important to notice that for example with the first variable, the distribution of normal samples continues quite high on the x-axis even though the curve goes very low on the y-axis. This means that the actual non-overlapping part of abnormal samples is quite small.

To have clearer idea of what variables divide the samples as normal and abnormal, a decision tree model is trained with the samples. Decision trees are hierarchical supervised models, used to for example classification.(Alpaydin, 2010) In this case supervised learning can be utilized because the predicted labels are available from the anomaly detection process. Decision trees provide results that can be very easily interpreted and for that reason are very popular (Alpaydin, 2010) The results of the model are visualized in the figure 19 below.

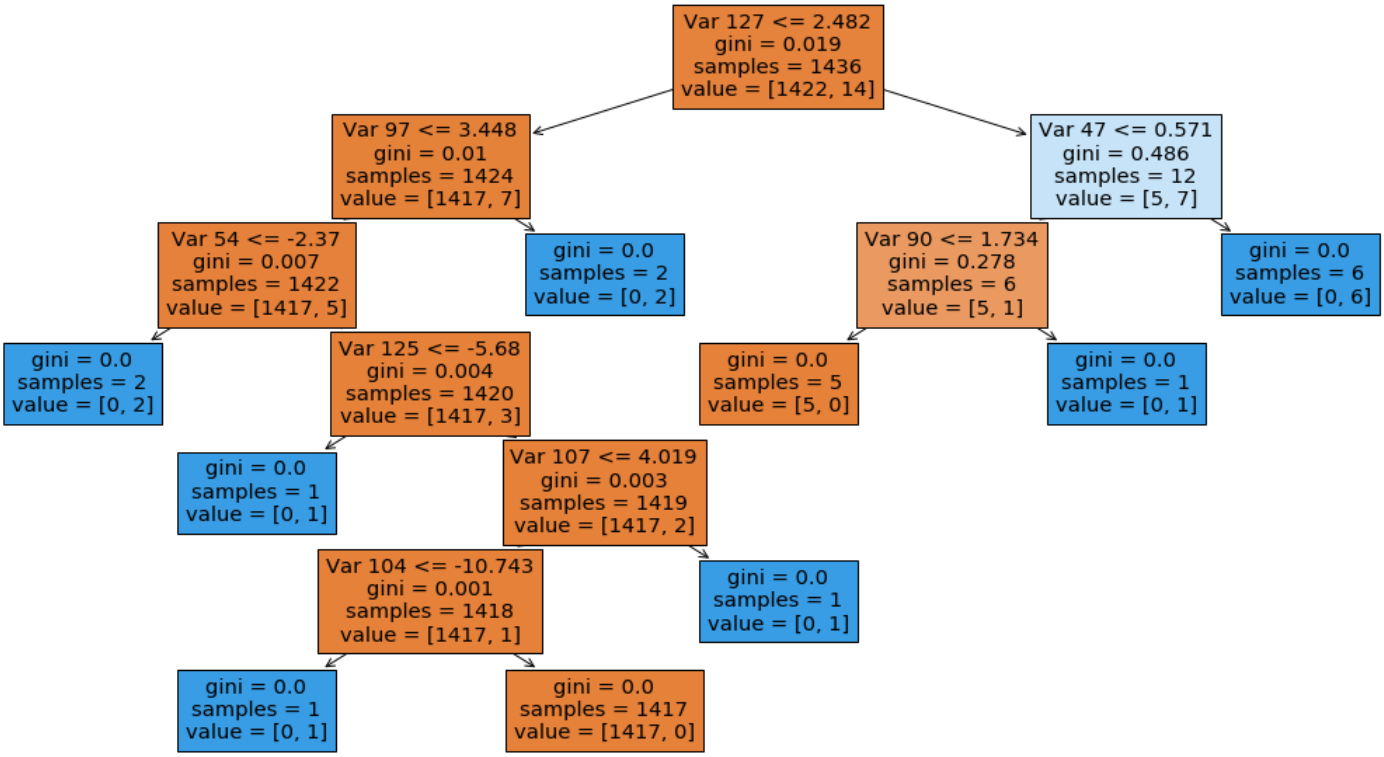

Figure 19 Results of the decision tree model trained

The colors in the figure 19 indicates on which class each node mostly consists of. The orange nodes indicate that the node consists mainly on normal samples and the blue ones represent abnormal samples. The number of each samples in each node can be seen between the square brackets. The first value is the number of normal samples. The gini value describes the purity of each node, meaning that how likely it is to incorrectly label a sample. The value can be between one to zero and the higher the value the higher the chance of misclassification is.

(Rebala, Ravi and Churiwala, 2019) This impurity is also illustrated by the shade of the color in the node, in terms of lighter the color the more equally the both classes are represented in the node.

Again, this method does not provide a clear answer to the issue that what causes these samples to be deemed abnormal, but it provides some useful information by illustrating where the unsupervised methods based the decisions to deem the samples abnormal. The first row of the nodes shows the name of the variable and the limit value on where the decision is made to divide the samples. The first division is very interesting because even it divides the abnormal samples 50/50 it only assigns five of the normal samples to the right side of the tree together with the seven abnormal samples. Also, the next division done divides the samples very well and the node ends up with the highest number of only abnormal samples in the whole tree with just two steps. This means that the variables Var 127 and Var 47 can be considered as one of the most differentiating variables in the dataset. The other steps on the right side and for all of the steps in the left side for the other seven abnormal samples in the tree conclude in nodes where the number of abnormal samples are one or two, so no reliable conclusions can be drawn from the other variables with this dataset.

7 CONCLUSIONS

In this chapter the findings from the literature review and case work are summarized and conclusions are made. This chapter also reflects on how well the research questions are answered and what can be done with the results. In the end further work and possible improvements are discussed.

7.1 Summary of results

The literature research showed that detecting anomalies in data is not a new subject and use cases vary between industries. Anomalies can be detected in a supervised manner or unsupervised manners depending whether the class labels are available. The supervised methods heavily rely on the neural networks, whereas unsupervised methods lean towards clustering algorithms. The literature also mentions algorithms and methods specifically developed to detect outliers in the data. However, these methods are not always the best performing depending on the use case.

The unsupervised anomaly detection literature is found to clearly divide into two segments. The first focuses on how to engineer the features in a way that the abnormal samples and values separate more clearly. These dimensionality reduction and feature extraction methods focused mostly on different variations of principal component analysis and autoencoders. The reason for using these modified methods usually comes from some very specific use case. The general PCA is selected to be used in this thesis because of its robustness compared to the modified versions.

The other segment of anomaly detection focuses on the actual methods on how to distinguish the abnormal samples. The most common methods in this area are one-class support vector machines, different clustering algorithms including hierarchical and density-based algorithms, and statistics based on PCA transformation. The literature did not bring up any unambiguously superior method for anomaly detection, but for example the OCSVM algorithm is used as a benchmark in multiple researches. Research on OCSVMs also provided multiple variations of the algorithm to make it more robust. In clustering algorithms, the HDBSCAN and gaussian

mixture models were one of the most used. The PCA based methods are based on how well the samples fit to the transformation done and for this purpose Hotellings T^2 and Q-residual statistics are used. These two methods can be stated as quite simple compared to the algorithms, but they are still widely used and found effective. Overall the literature review achieves to answer the research question presented regarding the literature and gives multiple examples of anomaly detection in industrial context.

For the actual case of detecting abnormalities from the testing data of product Alpha in total of four methods are used and PCA is selected for preprocessing. The PCA transformation is done also because of the Hotellings T^2 and Q-residual statistics are used. The other two methods are OCSVM and HDBCAN. All of these methods are used collectively to select the abnormal samples. As a final result 14 of the 1436 samples were deemed as abnormal and selected for further analysis, which is consistent with the expected rate of faulty units. The further analysis reveals variables that differ between the normal and abnormal samples. The differences in these variables are not as clear as it would be hoped for, since majority of the values that occur in the normal cases can also occur in the abnormal cases. Some information can still be gained from the analysis for example by using a decision tree. The decision tree model reveals variables that are causing the samples to be deemed abnormal. Almost half of the samples deemed abnormal can be se found just by looking at values of two variables. This also shows that there are multiple different types of anomalies detected and not every sample is caused by the same variable.

The research question regarding on detecting early failures of product Alpha remains partly unanswered in the time of the writing of the thesis. The results show that there are anomalous samples found in the data, but the implication to early failures in field use cannot be validated although the number of abnormal samples correspond to the expected field failure rate. The data to validate the results should be available within a year of the writing of this thesis.

As the final result of this thesis the actual tool to detect the abnormalities is provided for ABB to test and possibly implement as a part of their production line. The tool is provided as a Python program which includes the pretrained models for anomaly detection. The tool takes production testing data as an input and gives results whether the samples are deemed abnormal or not. These results can then be further analyzed in a way seen most suitable at the moment.

7.2 Discussion

The main outcome this thesis results is a tool that can be used to detect possible faulty units that pass the product testing. The testing data is used as the input and the tool provides a suggestion that something can be wrong with the unit in hand. The tool only provides a suggestion that the unit should be further examined. The decision on whether further analysis is carried out, the unit is retested or otherwise examined, is done by the experts in production testing.

This thesis also provided knowledge on the units tested previously and what variables cause variation between units. The data available in the time of the writing of this thesis couldn't be used to validate the results but if failures start to happen the units can be compared to the results. Also, if failures happen, and they don't match the units in the results, then conclusions on the suitability of this kind of methods on this kind of data can be drawn. Either way, the methods show that a group of samples deemed abnormal can be identified from the data, even if they don't necessarily indicate early failures in the field use of the product.

The validation could have been done in some level by analyzing the latest products that have come through the testing process and to see if the detected abnormal units really have some issues. This unfortunately could not have been done due the restrictions caused by the COVID-19 situation occurring during the writing of the thesis. Overall spending more time on the actual site of production and testing could have produced better results.

7.3 Further work

The next steps to continue the work done in this thesis are to wait for the possible failures to obtain labelled data and see if they match with the results presented and also to use the tool with new units tested. This can be considered as the most important thing to do considering to gaining concrete business value from this thesis.

The methods used for to analyze the testing data can also be used to analyze the data collected from the units when used in the customer sites. This means that the research done in this thesis

can be used as a basis of a predictive maintenance application if needed, because also with predictive maintenance the idea is to find out if something abnormal is happening during operation. Providing preventive maintenance as a service for customers could potentially bring new business for ABB.

Other things to consider in future are looking into other algorithms or methods to improve the current tool created during this thesis. Possibly the biggest improvement could come form moving into supervised methods, ss the data from early failures are collected. This would also mean that the testing process and attributes measured should remain the same in the future.

When more data is collected and the results can be validated, it is also possible that the current methods are deemed somewhat unsuitable. In that case when more data is collected the results can point towards possible improvements to the current methods or spark ideas for other methods. Because the current testing eliminates most of the failures it could be wise to add more attributes or change the way of measuring. Currently the measures are collected from a single point of time. For example, collecting the data as a time series could provide more opportunities to detect abnormalities.

For the algorithms used possible further research could focus on examining the different types of modified algorithms and to see if they yield better results. As mentioned before, different modifications to OCSVM and PCA are very common. Also, a possibility would be to make modifications to respond to the specific use case. This would of course need more research on algorithm development.

REFERENCES

ABB (2020a) *ABB in Finland.* Available at: https://new.abb.com/fi/abb-lyhyesti/suomessa (Accessed: 15.2.2020).

ABB (2020b) *About ABB.* Available at: https://new.abb.com/about (Accessed: 15.2.2020).

Alam, S., Sonbhadra, S.K., Agarwal, S. and Nagabhushan, P. (2020) *One-class support vector classifiers: A survey*. Knowledge-based systems, 2020-05-21, Vol.196, p.105754

Alpaydin, E. (2010) *Introduction to Machine Learning.* 2nd edn.The MIT Press.

Amer, M., Goldstein, M. and Abdennadher, S. (2013) *Enhancing One-Class Support Vector Machines for Unsupervised Anomaly Detection.* , Chicago, IllinoisNew York, NY, USA: Association for Computing Machinery, .

Amruthnath, N. and Gupta, T. (2018) *A research study on unsupervised machine learning algorithms for early fault detection in predictive maintenance.* pp. 355. 2018 5th International Conference on Industrial Engineering and Applications (ICIEA)

Angelopoulos, A., Michailidis, E.T., Nomikos, N., Trakadas, P., Hatziefremidis, A., Voliotis, S. and Zahariadis, T. (2019) 'Tackling Faults in the Industry 4.0 Era-A Survey of Machine-Learning Solutions and Key Aspects', *Sensors (Basel, Switzerland)*, 20(1), pp. 109. doi: 10.3390/s20010109.

ASQ (2020a) *THE HISTORY OF QUALITY.* Available at: https://asq.org/quality-resources/history-of-quality (Accessed: 11.4.2020).

ASQ (2020b) *WHAT IS STATISTICAL PROCESS CONTROL?* Available at: https://asq.org/quality-resources/statistical-process-control (Accessed: 11.4.2020).

ASQ (2012) *QUALITY ASSURANCE & QUALITY CONTROL.* Available at: https://asq.org/quality-resources/quality-assurance-vs-control (Accessed: 9.6.2020).

Bansal, Gaur and Singh (2016) *Outlier Detection: Applications and techniques in Data Mining*. 2016 6th International Conference - Cloud System and Big Data Engineering. pp. 373-377

Bolón-Canedo, V., Alonso-Betanzos, A. and Sánchez-Maroño, N. (2015) *Feature Selection for High-Dimensional Data.* Cham: Springer International Publishing.

Brandon-Jones, A., Slack, N. and Johnson, R. (2013) *Operations Management.* 7th, edn. United Kingdom: Pearson Prentice Hall.

Breunig, M., Kriegel, H., Ng, R. and Sander, J. (2000) 'LOF: Identifying density-based local outliers', *Sigmod Record*, 29(2), pp. 93-104.

Bro, R. and Smilde, A.K. (2014) 'Principal component analysis', *Analytical Methods*, 6(9), pp. 2812-2831. doi: 10.1039/C3AY41907J.

Campello, Ricardo J. G. B., Moulavi, D. and Sander, J. (2013) *Density-Based Clustering Based on Hierarchical Density Estimates.* . Mining. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 160.

Campello, Ricardo J. G. B., Moulavi, D., Zimek, A. and Sander, J.o. (2015) 'Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection', *ACM Trans.Knowl.Discov.Data*, 10(1). doi: 10.1145/2733381.

Capgemini (2019) *World Quality Report 2019-20.* Available at: https://www.capgemini.com/research/world-quality-report-2019/ (Accessed: 10.4.2020).

Chen, H., Jiang, B., Lu, N. and Mao, Z. (2018) 'Deep PCA Based Real-Time Incipient Fault Detection and Diagnosis Methodology for Electrical Drive in High-Speed Trains', *IEEE Transactions on Vehicular Technology*, PP, pp. 1. doi: 10.1109/TVT.2018.2818538.

Cheng, F., Raghavan, A., Jung, D., Sasaki, Y. and Tajika, Y. (2019) *High-Accuracy Unsupervised Fault Detection of Industrial Robots Using Current Signal Analysis.* . 2019 IEEE International Conference on Prognostics and Health Management (ICPHM), San Francisco, CA, USA June. pp. 1-8.

Deng, X. and Tian, X. (2015) 'Multiple Component Analysis and Its Application in Process Monitoring With Prior Fault Data', *IFAC-PapersOnLine; 9th IFAC Symposium on Fault Detection, Supervision andSafety for Technical Processes SAFEPROCESS 2015*, 48(21), pp. 1383-1388. doi: https://doi.org/10.1016/j.ifacol.2015.09.718.

Domingues, R., Filippone, M., Michiardi, P. and Zouaoui, J. (2018) *A comparative evaluation of outlier detection algorithms: Experiments and analyses*. Pattern Recognition Volume 74. pp.406-421

Doraisamy, S., Golzari, S., Norowi, N., Sulaiman, m.n. and Udzir, N. (2008) *A Study on Feature Selection and Classification Techniques for Automatic Genre Classification of Traditional Malay Music.*

Duda, R.O., Hart, P.E. and Stork, D.G. (2012) *Pattern Classification.* 2nd ed. edn. US: Wiley-Interscience.

Erboz, G. (2017) *How To Define Industry 4.0: Main Pillars Of Industry 4.0.*

García, S., Herrera, F. and Luengo, J. (2015) *Data Preprocessing in Data Mining.* Cham: Springer International Publishing.

Guo, K., Liu, D., Peng, Y. and Peng, X. (2018) *Data-Driven Anomaly Detection Using OCSVM with Boundary Optimzation. 2018 Prognostics and System Health Management Conference (PHM-Chongqing).* pp. 244-248

ISO (2015) *ISO 9000.* Available at: https://www.iso.org/standard/45481.html (Accessed: 14.11.2020).

Kacprzyk, J., Kacprzyk, J., Gunn, S., Guyon, I., Nikravesh, M. and Zadeh, L.A. (2006) *Feature Extraction : Foundations and Applications.* Berlin, Heidelberg: Springer Berlin Heidelberg.

Kaupp, L., Beez, U., Hülsmann, J. and Humm, B.G. (2019) *Outlier Detection in Temporal Spatial Log Data Using Autoencoder for Industry 4.0.* . Networks. Cham: Springer International Publishing, pp. 55.

Kawachi, Y., Koizumi, Y. and Harada, N. (2018) *Complementary Set Variational Autoencoder for Supervised Anomaly Detection. 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* pp. 2366 –2370

Kotler, P., Armstrong, G. and Opresnik, M.O. (2018) *Principles of marketing.* 17th edn. Harlow, England: Pearson.

Lienig, J. and Bruemmer, H. (2017) *Fundamentals of Electronic Systems Design.* Cham: Springer International Publishing.

Mack, D.L. , Biswas, G., Khorasgani, H., Mylaraswamy, D. and Bharadwaj, R. (2018) 'Combining expert knowledge and unsupervised learning techniques for anomaly detection in aircraft flight data', *at - Automatisierungstechnik,* 66(4), pp. 291-307. doi: 10.1515/auto-2017-0120.

Murphy, K.P. (2012) *Machine learning : a probabilistic perspective.* Cambridge, MA: MIT Press.

Rebala, G., Ravi, A. and Churiwala, S. (2019) *An Introduction to Machine Learning.* 1st ed. 2019. edn. Cham: Springer International Publishing.

Russell, E. and Chiang, L. (2000) 'Fault Detection in Industrial Processes Using Canonical Variate Analysis and Dynamic Principal Component Analysis', *Chemometrics and Intelligent Laboratory Systems,* 51, pp. 81-93. doi: 10.1016/S0169-7439(00)00058-7.

Schölkopf, B., Platt, J., Shawe-Taylor, J., Smola, A. and Williamson, R. (2001) 'Estimating Support of a High-Dimensional Distribution', *Neural computation,* 13, pp. 1443-1471. doi: 10.1162/089976601750264965.

Scikit-learn (2020) *Clustering.* Available at: https://scikit-learn.org/stable/modules/clustering.html# (Accessed: 3.6.2020).

Tax, D.M.J. and Duin, R.P.W. (2004) 'Support Vector Data Description', *Machine Learning,* 54(1), pp. 45-66. doi: 10.1023/B:MACH.0000008084.60811.49.

Vanem, E. and Brandsæter, A. (2019) 'Unsupervised anomaly detection based on clustering methods and sensor data on a marine diesel engine', *Journal of Marine Engineering & Technology*, , pp. 1-18. doi: 10.1080/20464177.2019.1633223.

Webster, J. and Watson, R.T. (2002) 'Analyzing the Past to Prepare for the Future: Writing a Literature Review', *MIS Quarterly*, 26(2), pp. xiii-xxiii.

Wise, B.M. and Gallagher, N.B. (1996) *The process chemometrics approach to process monitoring and fault detection*. Journal of process control. pp. 329–348

Yao, R., Liu, C., Zhang, L. and Peng, P. (2019) *Unsupervised Anomaly Detection Using Variational Auto-Encoder based Feature Extraction.* IEEE, pp. 1.