# PREDICTIVE MAINTENANCE OF CHROMATOGRAPHS

Lappeenranta–Lahti University of Technology LUT

Master`s thesis

2022

Diana Titova

Examiner(s): Professor Jan Stoklasa

Professor Tomáš Talášek

# ABSTRACT

Lappeenranta-Lahti University of Technology LUT

School of Business and Management

Business Administration

B.Sc. Titova Diana

**Predictive Maintenance of Chromatographs**

Master's Thesis

2022

73 pages, 27 figures, 17 tables, and 0 appendices

Examiners: Professor Jan Stoklasa and Professor Tomáš Talášek

Keywords: binary classification task, chromatographs, deep learning, machine learning, predictive maintenance, replacement

New methods of prognostic analytics for industries are invented constantly, and research of predictive maintenance has become widely popular in recent years, so it becomes a requirement for the company to implement data-driven decisions into the working process. There are many works considering predictive maintenance implementation. Nevertheless, input data highly influences the process of models implementation, and business cases devoted to this phenomenon are limited. This Master's Thesis provides an analysis of predictive maintenance models implementation in the biotechnological industry.

In the literature review, the main directions of predictive maintenance research and remaining useful life estimation topic were discussed. In the empirical part of the thesis methods of Machine Learning and Deep learning were applied and compared. Decision Tree Classifier and Random Forest Classifier outperformed other models in terms of four different metrics - accuracy, precision, recall, and f-score values – and training time. Since the company is interested in the interpretation of models predictions, the two models were chosen as the best algorithms for predictive maintenance with four metrics equal to 1 on both validation and test data. Based on the findings, managerial implications and theoretical contribution of the work are given.

# ACKNOWLEDGEMENTS

I would like to thank all the staff from LUT University, who helped me during my studies in Business Analytics Program, it was a great experience to study in Finland for a year. Also, I would like to express my special thanks of gratitude to my scientific supervisors Jan Stoklasa and Tomáš Talášek. Your valuable and reasonable comments truly helped to improve the quality of the thesis.

Finally, I would like to thank my father Pavel Titov and my mother Larisa Titova, and all my dear friends, who constantly supported me on my study journey.

Best regards,
Diana Titova

# List of abbreviations

PdM Predictive Maintenance

PM Preventive Maintenance

IoT Internet of things

ML Machine Learning

CNN Convolutional Neural Network

DL Deep Learning

LSTM Long short-term memory

KNN K-nearest neighbors algorithm

SVM Support-Vector Machine

RUL Remaining Useful Life

ReLU Rectified Linear Unit

# TABLE OF CONTENTS

1. Introduction ..... 8
1.1. Background ..... 8
1.1. Research questions ..... 9
1.2. Model implementation ..... 9
1.3. Structure of the thesis ..... 10
2. Literature review ..... 12
2.1. Main directions of predictive maintenance research ..... 12
2.2. Approaches to RUL estimation ..... 14
3. The mechanism of predictive maintenance and RUL estimation ..... 16
4. Context of the research ..... 18
4.1. Pharmaceutical companies' specifics of predictive maintenance implementation ..... 18
4.2. Company overview ..... 18
5. Methodology ..... 21
5.1. Machine Learning ..... 21
5.2. Classification models ..... 22
5.2.1. Logistic regression ..... 22
5.2.2. Decision Tree Classifier ..... 23
5.2.3. Random Forest Classifier ..... 23
5.2.4. Naïve Bayes Classifier ..... 24
5.2.5. SVM classifier ..... 24
5.2.6. KNN – classifier ..... 25
5.2.7. Deep Learning models ..... 26
5.3. Classification metrics ..... 28
5.4. Python Programming language ..... 29
6. Model application to Biocad`s data ..... 31
6.1. Biocad`s dataset ..... 32
6.2. Data preprocessing ..... 34
6.3. Results of data classification using ML models ..... 42
6.3.1. Prediction based on four-week interval ..... 43
6.3.2. Prediction based on two-week interval ..... 46
6.4. Results of data classification using DL models ..... 50
6.4.1. Prediction based on four-week interval ..... 50

6.4.2. Prediction based on two-week interval....56
6.5. Models comparison....61
7. Managerial application of the research....63
Conclusion....65
References....68

# LIST OF FIGURES

Figure 1. The exponential growth of the number of publications considering predictive maintenance. Image is taken from (Leohold, 2021)....13
Figure 2. Convolutional Neural Network. Image is adapted from (Beucher et al., 2022)....27
Figure 3. Recurrent Neural Network. Image is adapted from (Ogunmolu et al., 2016)....27
Figure 4. Number of observations from 10 chromatographs....34
Figure 5. Operating time of lamps....37
Figure 6. Correlation matrix....39
Figure 7. Feature importance by Extra Trees Classifier (4-week interval)....40
Figure 8. Feature importance by Decision Tree Classifier (4-week interval)....41
Figure 9. Classification results (visualization) based on the four-week interval (ML)....44
Figure 10. Decision rules based on four-week interval....45
Figure 11. Feature importance by Extra Trees Classifier (2-week interval)....47
Figure 12. Classification results (visualization) based on the two-week interval (ML)....49
Figure 13. Decision rules based on a two-week interval....49
Figure 14. ReLU function. Image is adapted from (Szandała, 2018)....51
Figure 15. CNN model prediction accuracy based on a four-week interval....52
Figure 16. Precision values based on the four-week interval (CNN)....53
Figure 17. Recall values based on the four-week interval (CNN)....53
Figure 18. LSTM-model prediction accuracy based on a four-week interval....55
Figure 19. Precision values based on the four-week interval (LSTM)....55
Figure 20. Recall values based on the four-week interval (LSTM)....56
Figure 21 CNN-model prediction accuracy based on a two-week interval....57
Figure 22. Precision values based on the two-week interval (CNN)....58
Figure 23. Recall values based on the two-week interval (CNN)....58
Figure 24. LSTM-model prediction accuracy based on a two-week interval....59
Figure 25. Recall values based on the two-week interval (LSTM)....60

Figure 26. Precision values based on the two-week interval (LSTM) .....60

Figure 27. (a) Chromatographs in work chart; (b) Number of operating lamps chart; (c) Operating time of lamps chart; (d) Need for maintenance chart .....64

LIST OF TABLES

Table 1. Confusion matrix .....28

Table 2. Python libraries used for ML and DL models implementation. .....30

Table 3. Example of the data provided by BIOCAD .....32

Table 4. Lamps replacement information .....34

Table 5. Information about detectors in the initial data provided by BIOCAD .....38

Table 6. Text extraction result .....38

Table 7. Detector label extraction result .....38

Table 8. Data preprocessing result .....42

Table 9. Cross-validation results based on the four-week interval (ML) .....43

Table 10. Classification results based on the four-week interval (ML) .....43

Table 11. Cross-validation results based on a two-week interval (ML) .....48

Table 12. Classification results based on the two-week interval (ML) .....48

Table 13. Cross-validation results based on the four-week interval (NN) .....52

Table 14. Cross-validation results based on the four-week interval (LSTM) .....54

Table 15. Cross-validation results based on the two-week interval (CNN) .....56

Table 16. Cross-validation results based on the two-week interval (LSTM) .....59

Table 17. The training time of the models .....62

# 1. Introduction

In this chapter the background of the research and general information considering the problem statement and research objectives are given.

## 1.1. Background

With the rapid development of modern technology, new methods to improve industry processes are being constantly developed. One of the widely used terms is Industry 4.0, which is also sometimes called Smart Industry. This concept can be divided into the following parts: smart manufacturing, Internet of Things (IoT), smart factory, and lights-out manufacturing (Sniderman et al., 2016). The idea behind the Industry 4.0 is the utilization of automation and modern data analysis models in the decision-making process in production facilities of large organizations. While industry 4.0 is about the implementation of data-driven automation and operations, predictive maintenance algorithms as an extension of condition-based monitoring make it possible to improve the maintenance process and productivity.

Current predictive maintenance (PdM) solutions can be divided into three major groups (Montero Jimenez et al., 2020):

1. Knowledge-based
2. Physics-based
3. Data-driven

Knowledge-based solutions are built on the assumption that experts and researchers that use the equipment can predict the need for maintenance. Even though an approach based on experts` knowledge is easy to use and implement, it often results in errors and delays in production. Physics-based solutions consider laws of physics to assess the degradation of the model and its components. The last solution is a data-driven one. It considers data getting from the devices continuously (Zhang et al., 2019).

With the development of machine learning (ML) algorithms and data analysis techniques data-driven solutions have increased in number (Tao et al., 2018). It is an obvious trend, predictive maintenance can help companies to anticipate problems before they happen, so that business

processes remain stable and profitable. Moreover, such decisions raise the profit margin and productivity of the company. For example, the PwC report discussed the influence of predictive maintenance implementation in factories: it reduces cost by 12 percent, improves uptime by 9 percent, reduces safety, health, environment, and quality risks by 14 percent, and extends the lifetime of an aging asset by 20 percent (PwC, 2018).

## 1.1. Research questions

The data for the thesis was provided by the biotechnology company BIOCAD. BIOCAD uses chromatographs to conduct analyses, each chromatograph has a lamp whose operating time influences the accuracy of the analysis result. The main task is to predict the need for the lamp replacement. Currently, the company`s maintenance management is based on a run-to-failure approach, when replacement takes place due to actual failure of the equipment which results in delays, financial losses, waste of resources due to loss of analyses results because of unnecessary replacements. It is not convenient to have spare lamps since a different number of lamps must be replaced each period, so an improvement of this approach can be the implementation of predictive maintenance algorithms to reduce maintenance costs and avoid unexpected failures.

Since delivery of the lamps takes about 2-4 weeks, lamps within the period of 4 or 2 weeks before the average usage period (3000 hours) are considered as close to failure. Therefore, predictions based on 4- and 2-weeks intervals were done. These two thresholds can also be used for the comparison of the models' accuracy. Four-weeks based prediction is a positive situation when the data is provided fully and the operating time of the lamps can be calculated accurately and precisely, while a two-week based prediction can be used as a pessimistic scenario when there is a data loss problem, these periods provide a confidence interval which can be used for model assessment.

The thesis is focused on the following research questions:

1. How well can machine learning and deep learning models classify the data in terms of accuracy, precision, and recall?
2. Are predictions of machine learning and deep learning models equally accurate or not?
3. Can machine learning and deep learning models attain the same level of prediction accuracy based on four-week and two-week interval before the expiration time of the lamp?

Machine learning (ML) algorithms can assimilate, aggregate, and synthesize production data to extract patterns from the data, over time models can be trained to respond to the new information, that is why the usage of ML models can strengthen PdM.

Machine learning algorithms provide effective tools for predictive maintenance implementation. A big advantage of the ML models is the ability to extract insights and patterns from the massive volume of data. The usage of Machine Learning algorithms will allow us to find answers to the proposed research questions. The data provided by BIOCAD includes time values and information from the equipment in textual format. Thus, an Ordinal Encoder is required to transform the text into numerical form to prepare the data for supervised ML models. Textual data includes important information considering the work of the equipment. Therefore, data transformation and extraction will be needed.

Formulation of the task in the form of the binary classification problem applies to the data and the company`s motives because binary classification predictions allow having easy to implement visual indications of the expected failure. Several ML and DL models can be used for predictive maintenance, some of them (SVM, KNN) are well-discussed (Nikfar et al., 2022; Zhao, 2017), others (Decision Trees, CNN) are topical (Yurek, 2019; Zhao, 2017), but all of them can produce meaningful results. Therefore, classification models, such as logistic regression, decision tree classifier, random forest classifier, k-nearest neighbor (KNN) algorithm, naïve Bayes, convolutional neural network (CNN), Long-Short Term Memory (LSTM) model will be used in the thesis. Mentioned models can help to find interrelations between the need for lamp replacement and other variables presented in the data.

## 1.2. Structure of the thesis

The first part of the thesis describes the implementation of a predictive maintenance algorithm. A literature review of major articles considering predictive maintenance and different approaches to the remaining useful life estimation is the second part of the thesis. Then the focus of the thesis is on the mechanism of predictive maintenance and specifics of predictive maintenance in biotechnology companies. The fifth chapter will explain the usage of ML and DL models in the thesis, the principle of computation, and distinctions in results interpretation. The thesis will further present supervised learning classification models results, and comparison analysis. Models

creation and execution are done via Python programming language. Then the overview of the dataset provided by Biocad, and the results of the application of the mentioned models will be given. Finally, the general conclusion on models applicability and model comparison results will be given.

# 2. Literature review

The literature used in the thesis can be divided into two main groups: main directions of predictive maintenance research and approaches to remaining useful life estimation.

## 2.1. Main directions of predictive maintenance research

Machinery maintenance can be divided into two groups: planned and unplanned maintenance. The former can be further divided into three other groups: predictive maintenance, preventive maintenance, and corrective maintenance. The latter is presented in two cases: emergencies, and breakdowns. Preventive Maintenance is usually carried out at predetermined intervals or according to prescribed criteria. Corrective Maintenance is carried out following the detection of an anomaly and is aimed at restoring normal operating conditions. Predictive maintenance uses predictions on when exactly the maintenance would be needed (Project Services Co., 2017).

Predictive maintenance includes several aspects and approaches. The theme is well-discussed, and many prognostic methods have been presented in academic papers that will be discussed below. According to the research ‘Prognostic Methods for Predictive Maintenance: A generalized Topology’, there is a rapid increase in the number of publications considering predictive maintenance that indicates a significant interest in the problem (Leohold, 2021). The trend looks as following (Figure 1).

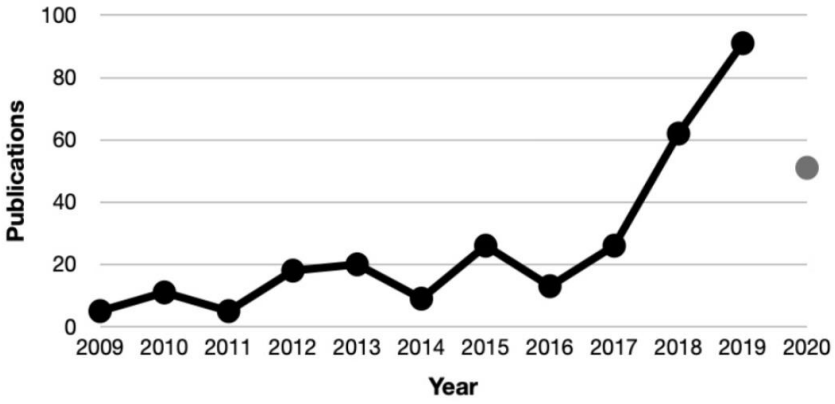


Figure 1. The exponential growth of the number of publications considering predictive maintenance. Image is taken from (Leohold, 2021).

Articles consider different stages of the predictive maintenance process, from production and manufacturing to models' application. For example, predictive maintenance approaches have been discussed in the research of Efthymiou et al. (2012), popularity of the PdM algorithms is obvious: implementation of PdM can save up to 60-75% of total costs of manufacturing activity.

There are many works considering the general idea of predictive maintenance. For example, there is a work by Wieger Tiddens which concerns the general theory of predictive maintenance, main models, frameworks and applications, and industry. The author concludes that there is a complexity of the application of PdM and a lack of theoretical guidance in selecting suitable techniques, identifying suitable candidates, and evaluating the added value of predictive maintenance application. (Tiddens, 2018). General data-driven approaches were also discussed in the article written by Zhang. It separated the implementation process into operational assessment, data acquisition, feature engineering, and modelling (Zhang et al., 2019).

Machine learning and deep learning models strengthen predictive maintenance implementation. ML is the process of model learning through the experience without being explicitly programmed. Deep learning is usually considered a subset of ML. The first difference between these two terms is the volume of data needed for training. ML performs well on small datasets, while DL models need large datasets to be trained properly. Consequently, DL requires significant computational resources for training. Besides, features must be identified manually in ML, while DL can learn features automatically (Wehle, 2017).

Successful application of ML and DL models for predictive maintenance is well described in the literature. Zhai et al. have used an unsupervised learning approach to cope with unlabelled data and received promising results (Zhai, 2021). Nikfar et al. have used a support vector machine, backpropagation neural network, and random forest to detect abnormal motor behaviour. All models resulted in high confidence in the prediction (Nikfar et al., 2022). LSTM-classifier was used and validated in a real application case study (Nguen et al., 2021). A comparison of machine learning and deep learning models in predictive maintenance was also discussed in several works. Comparative analysis of ML algorithms showed that Random Forest outperformed other machine learning models (Ayvaz, 2021). Tiago Zonta et al. (2022) compared the usage of RNN and DNN in maintenance prediction. Some articles include relevant code examples in R or Python (Chekkala, 2020; Mishra, 2018).

To sum up, the problem of predictive maintenance based on ML and DL approaches has significant interest and is well-studied. Several methods are analysed and compared in the literature. However, there are still several issues. One of these is that most models are not designed for and are not validated with industrial data (Zhai, 2021).

## 2.2. Approaches to RUL estimation

The key term included in the predictive maintenance discussion is remaining useful life (RUL) estimation, which involves the prediction of the period that a component or a device will be able to operate before warranting replacement. RUL problem discussion can be divided into several topics: knowledge-based models, physical models, data-driven models, and deep learning (Borst, 2020). Similar classification can be found in many academic articles (Kang et al., 2021). The first one considers decisions based on previous failures, while physical models try to model the process.

Deep learning models implementation for remaining useful life estimation is well-discussed in the literature. RUL estimation based on the recurrent neural network was proved to be accurate in the work of Wang et al (2020). There are works considering the use of LSTM-algorithms and their effectiveness in RUL estimation (Wu et al., 2020). A deep feature disentanglement transfer learning network has been effectively validated and proved to be effective for RUL prediction (Hu, 2022). An approach of using the multilayer perceptron neural network for remaining useful life estimation substantially enhanced predictive maintenance results (Kang, 2021).

However, the usage of ML models for the case of RUL prediction is not widely presented in the literature. Such ML models as Linear Regression, Bayesian Linear Regression, Poisson Regression, Neural Network Regression, Boosted Decision Tree Regression, and Decision Forest Regression were used to predict the remaining useful life (Yurek, 2019). Comparison of ML and DL models based on the industry data was not the focus of many works. A comparative method that was not actively applied to this matter may result in a discovery of hidden data patterns that will lead to further managerial conclusions which would be beneficial for companies.

# 3. The mechanism of predictive maintenance and RUL estimation

Predictive maintenance algorithms monitor and collect data, compare it with conditional baselines and detect any abnormalities or specific parameters. If there is a trigger, PdM models show that there is a required repair.

This prediction model is useful for manufacturers because companies can anticipate when maintenance is needed, so they will not need to fix unexpected damages. There are several benefits of predictive maintenance implementation: it can reduce or even eliminate unscheduled equipment downtime, increase production capacity, reduce maintenance costs, and increase equipment lifespan (Carvalho, 2019). For these reasons, predictive maintenance is now widely used in many spheres, and companies are actively investing in the development of these prognostic models.

There are several types of predictive maintenance implementation in the industry:

a. Maintenance Planning and Scheduling. The goal of these algorithms is to predict maintenance considering company specificity and policy.

b. Reliability- and Degradation- based Decision Making. This model differs in the implementation of degradation rates to minimize long-term costs.

c. Joint Optimization. This area also considers production and supply objectives.

d. Multi-State and Multi-Component Systems Optimization. This approach influences intermediate decision-making through the identification of intermediate stages.

e. Maintenance Cost and Risk Estimation and Optimization. The focus of these models is the facilitation of the decision-making process by considering risk and costs (Alexandros Bousdekis et al., 2019).

Implementation of predictive maintenance requires the following steps:

1. Data collection
2. Fault detection
3. Machine fault diagnostics
4. Machine prognostics
5. Deciding on scheduling or resource optimization (K.S. Jardine et al., 2006)

The remaining useful life estimation, which is a way to strengthen predictive maintenance predictions, can be calculated using different approaches. The method of calculation is usually chosen based on the data available, three approaches can be mentioned:

1. Usage of the lifetime data that can show the period before the failure of similar machines
2. Run-to-failure history of similar machines. (Run-to-failure is a maintenance strategy that assumes that replacement is taken place only when the equipment has failed (Keyur, 2020))
3. A known threshold that can be used as an indicator to detect the need for replacement (Mathworks, 2018)

In the thesis, the third approach was applied, as it is stated by the necessity to order and replace lamps within four-week and two-week intervals.

# 4. Context of the research

In this chapter, the context of the research, new challenges for pharmaceutical companies, and the effect of digitalized technology are discussed.

## 4.1. Pharmaceutical companies' specifics of predictive maintenance implementation

In recent years the pharmaceutical industry has experienced major changes in terms of the decision-making process, management decisions, data analysis, and customer demand. For sure, this industry is one of the most growing global sectors with sales of more than 1228.45 billion dollars in 2020 (Mikulic, 2021). The annual rate of the pharmaceutical market is about 6% since 2017 (Dossier, 2018).

Unexpected failures lead to loss of analysis results, and delays in their delivery, moreover, they can cost a company a huge share of the revenue, so pharmaceutical firms look for any approaches to optimize operating costs and increase profitability. Consequently, there is a strong uptrend in the number of predictive maintenance solutions. For example, the North America PdM healthcare market has increased almost three times from 2017 to 2022 (from 51.9 to 151.5 million dollars) (Otto, 2019). During the COVID-19 pandemic pharmaceutical companies faced new challenges, so digitalized technology is getting more and more important because it improves performance and creates more accurate planning and forecasting.

## 4.2. Company overview

BIOCAD is an international biotechnological company. The firm has been developing, researching, and manufacturing medicines for over 20 years. The company operates worldwide and has representative offices in the United Arab Emirates, China, Brazil, and Vietnam. It is famous for effective, safe, and affordable drug supply solutions. Biocad`s mission is to improve and extend people's lives through the provision of effective, safe, and affordable integrated drug

supply solutions. The company carries out a full cycle of development of new drugs in its research centers and laboratories (BIOCAD, 2022)

BIOCAD implements structure-based drug design technology, they use modern computer modeling methods. The company has three R&D centers that are engaged in the creation of both original and generic drugs. The company launches research programs: MabNext(Creation of innovative drugs based on monoclonal antibodies for the treatment of oncological and autoimmune diseases), ChemNext(Obtaining of low-molecular compounds for the creation of revolutionary drugs of a chemical nature), and GeneNext (An advanced project for the creation of in vivo and ex vivo gene therapy aimed at developing a fundamentally new class of drugs). In 2019 BIOCAD invested 2.038 billion roubles in the development of original drugs (BIOCAD, 2019-2020).

In pharmaceutical biotechnologies, the main work is carried out with proteins: these substances affect any biological process in the body. Depending on the structure of the molecule, a protein can treat a particular disease. There are many natural proteins, but the main goal is to find the right molecule for the treatment of the disease, it might be a tedious task. To save time scientists in BIOCAD create and use algorithms that help find the right molecule according to the given parameters. It is essential to conduct virtual experiments because they are important for the early stage of drug development. They allow researchers to save time and resources when looking for optimal solutions. Therefore, BIOCAD tries to automate various processes that are usually carried out manually, for example, filling out documentation and passports for drugs, and accounting for laboratory objects (RBK Group, 2021).

In production, BIOCAD staff uses different devices to conduct analyses and experiments. One of the devices is a chromatograph. There are more than 100 chromatographs in use, they have various components and different frequencies of work. Chromatographs use two types of lamps and there is a need for their regular maintenance. All lamps have an RFID tag with operating time. Replacement takes place now according to the actual failure of the lamp (the lamp simply does not ignite). The lamp may also be replaced if the failure is detected by the researcher. The cost of each lamp is 1200 EUR, lamp limit of usage is about 3000 hours. It is not possible for the company to have spare lamps for all occasions, because a different number of lamps must be replaced each month, it is not convenient and profitable to order them in advance. The process of maintenance is not automatic, which leads to termination of work, because the time to order and deliver the lamp takes from 2 to 4 weeks, therefore there might be replacement delay, which results in suspension of studies. More precisely, if the lamp is replaced during the analysis it takes about 4

hours to replace it and the results of the research are not significant so the whole process should be repeated, if replacement is conducted when the analysis is finished, it takes only 30 minutes, and no results are lost.

According to the information provided by the company, on average, the contribution of the predicted maintenance of 20 processes of lamps replacements per year will save 65 hours of employers' working time and 20 preserved research results. Moreover, since the average number of lamps used per year is 20 and the price of the lamp is 1200 euros, lamps delivery costs 24 000 euros per year. Therefore on-time maintenance and lack of necessity to deliver unneeded lamps will save the company money.

All in all, BIOCAD is an established pharmaceutical industry player that has extensive knowledge and data related to biotechnology.

# 5. Methodology

In this chapter applicability of machine learning and deep learning models for predictive maintenance is discussed. Classification metrics used for models assessment are explained.

## 5.1. Machine Learning

Machine learning applications can be used in different spheres – retail (market basket analysis, time patterns analysis), banking (card fraud detection, customer segmentation), telecommunications (customer loyalty analysis), insurance (insurance fraud detection, risk analysis), molecular biology and genetics (drug design). Machine learning algorithms can be separated into two groups: supervised and unsupervised learning. They are different in a basic principle of work -while supervised algorithms use already known labels to train and test models, unsupervised – train models without any labels in the training dataset (Alloghani et al., 2020).

Supervised ML algorithms use the predetermined label in addition to the use of input variables (Kotsiantis, 2007). These models try to classify (classification task) or predict (regression task) the output attribute while considering different performance measures (Mathworks, 2016). Examples of supervised algorithms are logistic regression, Lasso and ridge regression, decision trees classifier, random forest classifier, and SVM (Support Vector Machine) classifier (Dangeti, 2017).

To prevent the occurrence of such problems as data leakage or overfitting, it is preferable to split the data into three parts – train, validation, and test data. The former is used to construct contingent functions for the classification or prediction of the output variable.

To be sure, that the models can be used in production, their predictions based on the unseen data should be assessed. To evaluate the model, cross-validation techniques are often used. One of the most popular validation techniques is k-fold cross-validation. It splits the data into k folds, the model trains on the first k-1 folds, while the remaining k folds are used as the test set. The process is repeated using all folds, and then the result is calculated as the mean of all models. Usually, k is equal to 10, as it was shown to produce accurate predictions on different datasets and models. However, mentioned validation technique can be misleading if the data is imbalanced. A modification of k-fold cross-validation, which is called stratified k-fold cross-validation, can be used to preserve class distribution in each fold (He, 2013).

## 5.2. Classification models

Classifications models for the thesis were chosen based on their applicability to predictive maintenance tasks. There are classical and topical methods for predictive maintenance implementation. Such models as RNN, CNN, SVM, Decision Tree, and Random Forest proved to provide accurate results within predictive maintenance and remaining useful life estimation areas (Naik, 2017; Wu, 2017; Zhao, 2017). Topical methods such as usage of Logistic Regression and KNN-classifier were not analysed based on the industry data. Their prediction accuracy varies based on the data quality and problem specification. Therefore, a comparison of classical and topical models was conducted. To get a deep understanding of methods, information considering the principle of calculations and work of the following models was analysed.

### 5.2.1. Logistic regression

Logistic regression is an algorithm that models binary dependent variables. The model compares two probability values from output and input variables and classifies the data based on the threshold. The logistic regression model can be problematic to use in terms of an imbalanced dataset, in these cases changes in threshold might be needed.

Mathematically, logistic regression models a binary outcome which can have two possible values (1 or 0). It uses a sigmoid function to return the probability value. The model is widely used for the binary classification task. A general formula for logistic regression is of the following form:

$$p(x) = \frac{1}{1 + e^{-(x-\mu)/s}} \quad (1)$$

where μ represents the midpoint of the sigmoid function – location parameter, $s$ is a scale parameter (Tolles et al., 2016).

Logistic regression is a simple model to tune because it does not have many parameters. The threshold and regularization type can be chosen to tune the model. Since it can be prone to misleading results in imbalanced classification tasks, hyperparameter tuning can be used.

The logistic regression model proved that it can be used for predictive maintenance, though the discussion is limited (Faraj, 2021). Although the model is simple and straightforward, it can be used as a baseline or to understand a general picture of the data (Staples, 2018).

### 5.2.2. Decision Tree Classifier

Decision Tree Classifier is a widely used algorithm for classification, it is famous for easily interpretable results. Decision Tree Classifier has internal nodes which represent features of the data, and the branches represent the decision rules, they are developed by partitioning the dataset into subsets based on the most important features. Final calculations – classification – are conducted at the so-called leaves of the model.

Decision Tree algorithms can be prone to overfitting, therefore thorough selection of the parameters, cross-validation techniques, or pruning are needed. There are two measures of impurity in the node: entropy and Gini; entropy provides the amount of impurity or uncertainty of the attributes. It is calculated using the following formula:

$$I = -\sum_{i=1}^{N} \mathrm{p}_i * \log_2(\mathrm{p}_i) \quad (2)$$

where $p_i$ is the proportion of class that belongs to the class N of the node.

Gini calculates the probability of a feature being classified incorrectly when selected randomly, it is calculated using the following formula:

$$\text{Gini} = 1 - \sum_j p_j^2 \quad (3)$$

where $p_j$ is the proportion of class that belongs to the class N of the node (Capozzoli et al., 2016).

The selection of one of the measures can influence the model results, therefore the influence of hyperparameters should be assessed.

### 5.2.3. Random Forest Classifier

The principle of Random Forest classifier is built on the combination of decision trees. It is a widely used ensemble method that aggregates results from several predictors.

While training, the algorithm uses such approaches as bootstrapping to include randomness into the calculations by splitting each tree node into subsamples. Therefore, with the addition of other trees, the model does not overfit, which is a huge advantage of the method. At the end of

calculations, the algorithm aggregates the predictions from all decision trees. This algorithm is better for generalization, but less interpretable due to a bigger number of parameters in comparison with Decision Tree algorithms (Naik, 2017).

The Random Forest model was used for predictive maintenance in the comparative study of machine learning algorithms, the results have shown that the RFs made predictions more accurate than Neural Network and Support Vector Regression models (Wu, 2017).

### 5.2.4. Naïve Bayes Classifier

Model is built on principles of Bayes Theorem, which considers the probability that an event will happen given that another event has happened:

$$P(A/B) = \frac{P(B/A)*P(A)}{P(B)} \quad (4)$$

where P(A/B) is a probability of A occurring given the evidence that B has occurred, P(B/A) is a probability of B occurring given the evidence that A has occurred, P(A) is a probability of A occurring, P(B) is a probability of B occurring.

Naïve Bayes Classifier is a semi-supervised algorithm because it can be used in both clustering (unsupervised problem) and classification (supervised) tasks (Mossotto et al., 2017). Naïve Bayes Classifier creates a Bayesian network, which is generated by outcome conditional probability which is based on probability given by input variables (Qiu et al., 2016).

The big advantage of the Naïve Bayes classifier is that it does not need much data to be trained properly. While the Naïve Bayes classifier is usually used for text classification tasks or problems with multiple classes, it can be used as a good baseline model for comparison with other algorithms. Usage of Naïve Bayes Classifier for RUL estimation was discussed in the work by Ng et al., where its accuracy was compared with that of Support Vector Machine. Naïve Bayes Classifier generated competitive prediction performance over the SVM (Ng, 2014).

### 5.2.5. SVM classifier

Support-Vector Machine classifier uses margins to classify the data. The model considers the position of the object base on the position in relation to the border. Each data item is represented as a point in n-dimensional space, while margins are defined as the distance between two

supporting vectors separated by a hyperplane. The model tries to minimize possible classification errors and maximize the distance between each class. Hyperplane categorizes points into classes. A big disadvantage of Support Vector Machine classifiers is that they are susceptible to overfitting, therefore results should be considered thoroughly (Rafiei et al., 2017).

There are several advantages of using an SVM classifier. First, it has good generalization capability which helps to prevent the model from the overfitting problem. Support Vector Machine can efficiently handle non-linear data. SVM classifier is comparatively stable because minor changes in data do not affect the hyperplane greatly. The model showed great results in handling sequential data, for example, one research showed that SVM even outperformed Artificial Neural Network for time-series forecasting (Samsudin, 2010). It also presented promising results in RUL prediction of the rolling bearing and the general applicability of the model for any prognostic task was concluded (Roulias, 2013).

#### 5.2.6. KNN – classifier

KNN classifier is a widely used algorithm for a classification task. It has the following advantages: researcher can choose distance measure metric which gives additional flexibility, results of the classification are getting better with adding more data to the training data. However, it should be mentioned that KNN is very sensitive to outliers, model implementation can take relatively more time as data grows (Altman et al., 1992).

One of the most important parameters is K – it refers to the number of nearest neighbors to the data point. This is a crucial variable that influences the whole decision-making process and division into classes. Several distance metrics can be used in KNN. The most widely used is Euclidean Distance, which is also called L2 Norm (Cohen, 2004).

$$\text{Euclidean Distance} = \sqrt{(x_2 - x_1)^2 - (y_2 - y_1)^2} \quad (5)$$

Manhattan distance is the distance between two points measured along axes at right angles. In a plane with $p_1$ at $(x_1, y_1)$ and $p_2$ at $(x_2, y_2)$, the formula is the following (Black, 2019):

$$\text{Manhattan distance} = |x_1 - x_2| + |y_1 - y_2| \quad (6)$$

Another one is Hamming distance is the number of bits that differ between two binary strings. More formally, the distance between two strings A and B. Distance metric is important for KNN

classifier, it helps to train data more efficiently. There is a possibility to get better results by using different instance metrics (Sharma, 2019).

### 5.2.7. Deep Learning models

Deep learning is usually considered as a subset of machine learning because it generally uses the same principles, but it differs in the architecture of the algorithms: there are input, hidden, output layers that construct the network scheme. Each hidden layer uses one of the various mathematical functions to calculate the output and transfer it forward. The data is analysed consecutively which is useful for many machine learning models.

Deep learning models iterate many times till a certain metric is achieved. Moreover, there is a tunable parameter – epoch – which helps to improve the model result by deciding how many times the entire dataset will be passed forward and backward through the network. There are two stages in deep learning models training: forward feature abstraction and backward error feedback (Goodfellow, 2016).

CNN and RNN models' usage was proved to be applicable to predictive maintenance and remaining useful life estimation tasks (Zhao, 2017).

Convolutional neural networks use linear operation between matrixes named convolution. CNN uses non-linearity within hidden layers. Another important thing – convolutional neural networks use a backpropagation algorithm to reduce the error while training the model. While using CNN as a classifier, SoftMax or sigmoid function should be used at the last layer – it classifies the data at the last stage (Konda et al., 2019).

Softmax is a generalization of the already mentioned logistic function which normalizes the network`s output to the probability distribution (Goodfellow et al., 2016). The Softmax function is presented by the following formula:

$$s(z_i) = \frac{e^{z_i}}{\sum^n e^{x_j}} \tag{7}$$

where $z_i$ is an input vector, $e^{z_i}$ is a standard exponential function for input vector, n is a number of classes in the multi-class classifier, $e^{x_j}$ is a standard exponential function for the output vector.

An example of the general structure of the Convolutional Neural Network model looks as following (Figure 2).

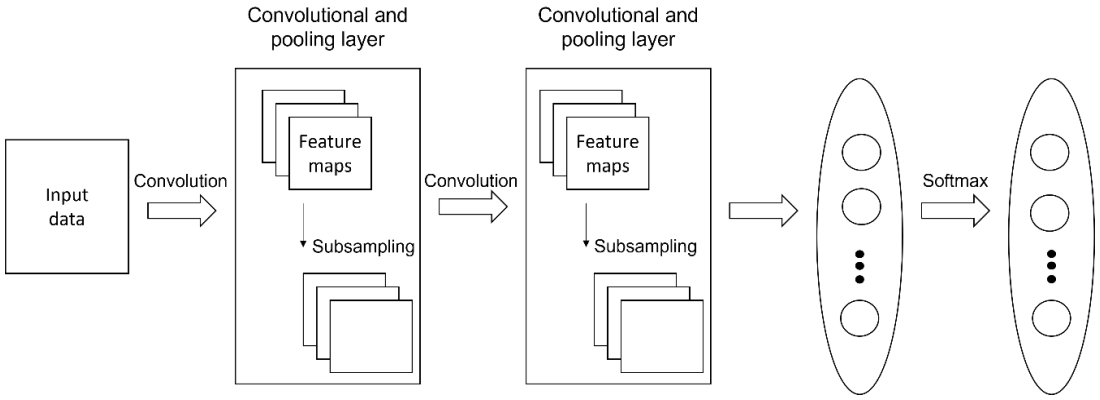


Figure 2. Convolutional Neural Network. Image is adapted from (Beucher et al., 2022).

Although Convolutional Neural Networks are highly popular for image classification, they can be used for other types of tasks.

Another type of neural network is a Recurrent Neural Network (RNN). It follows the same logic and structure but differs in one aspect – it considers a sequence of data values. It is highly used in the time-series analysis where dependencies matter. RNN allows the usage of previous outputs as new inputs (Sherstinsky, 2020). An example of the general structure of the Recurrent Neural Network model looks as following (Figure 3).

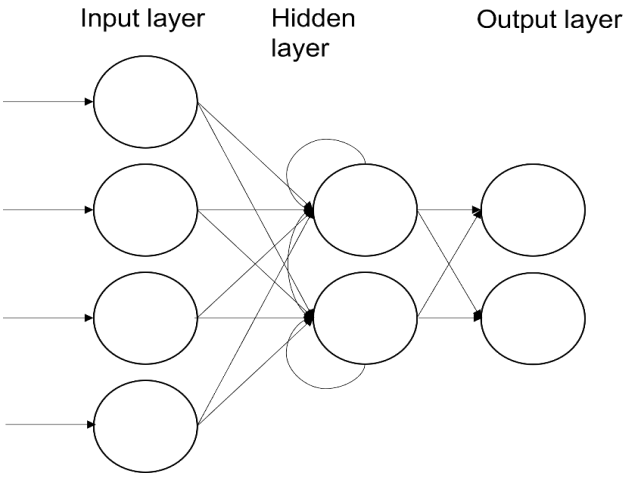


Figure 3. Recurrent Neural Network. Image is adapted from (Ogunmolu et al., 2016).

LSTM-model is a variant of the RNN model, which uses iterative algorithms to learn long-term dependencies. LSTM-model can extract information from the previous cell and transfer it forward.

It is highly used to overcome the vanishing gradient problem. LSTM scheme is presented by connected memory blocks (Hochreiter, 1997). LSTM-model applies to the data provided by BIOCAD because it can model complex multivariate data sequences

Since both ML and DL models can be used for BIOCAD`s data, they were compared in terms of the models' ability to conduct binary classification task. Two major differences are the following:

1. Convolutional Neural Networks are computationally cheaper than Recurrent Neural Networks because Recurrent Neural Networks can not use parallelization while classifying the data.

2. Convolutional Neural Networks can see the data from a broader perspective because Recurrent Neural Network learns the data only from the timestamp it needs to predict. (Hassan Ismail Fawaz et al., 2019)

## 5.3. Classification metrics

Results of classification are presented as a list of predicted labels – 0 and 1. To evaluate each model one should consider classification metrics which help to understand whether the model achieved a certain level of prognostic accuracy.

For the binary classification task, evaluation of the best solution can be done based on the confusion matrix, where a row of the table represents predicted by the model class, while the column represents an actual class. With the help of these instances, several terms are invented – true positive (number of positive instances that are correctly classified), true negative (number of negative instances that are correctly classified), false positive (number of misclassified positive instances), false negative (number of misclassified negative instances) (Hosin et al., 2015). The table looks as follows (Table 1).

Table 1. Confusion matrix

| | **Actual positive class** | **Actual negative class** |
|---|---|---|
| **Predicted positive class** | True positive | False positive |
| **Predicted negative class** | False negative | True negative |

False positive and false negative predictions can be equally bad or not depending on the data and the situation that is analysed. In the case of a predictive maintenance problem, both types of mistakes can worsen the working process of the company: while false positive result leads to waste of resources and extra costs, false negative result contributes to not only loss of money, but also to the interruptions in analysis and corrupted analysis results, which is a worse situation. Therefore, minimization of the false negative rate as a priority should be achieved.

To assess prediction accuracy and compare selected ML and DL models, the following metrics were chosen. One of the most used classification metrics is accuracy, it is based on the percentage of correct predictions over total class.

$$\text{Accuracy} = \frac{number\ of\ correct\ predictions}{total\ number\ of\ observations} = \frac{TP+TN}{TP+TN+FP+FN} \quad (8)$$

This metric is easy to compute and implement, though, despite its popularity among researchers, accuracy can be a highly biased metric in case of imbalanced classification, it is less favored towards minority class instances (Hosin et al., 2015).

Precision shows how many of the selected positive instances are positive. This metric is relevant to use in the cases where identification of true positive results is a priority (Herrera et al., 2016).

$$\text{Precision} = \frac{TP}{TP+FP} \quad (9)$$

A recall is used to measure the fraction of relevant observations selected (Hosin et al., 2015).

$$\text{Recall} = \frac{TP}{TP+FN} \quad (10)$$

In terms of classification metrics, precision and recall are often in tension, therefore it is seen as a trade-off between two metrics. Due to the presence of the imbalance in the data provided by BIOCAD and since both false negative and false positive results are bad for the company, F-measure is a good metric to assess prediction accuracy. F-Measure (F-mean) represents the harmonic mean between precision and recall values (Hosin et al., 2015).

$$\text{F} = \frac{2*precision*recall}{precision+recall} \quad (11)$$

## 5.1. Python Programming language

Mentioned Machine Learning tasks will be implemented in Python programming language with the usage of Jupyter Notebook. Python is considered as one of the best options for programming purposes due to the simplicity and consistency of the language (Thanaki, 2017). It allows to conduct data preprocessing, data analysis, train and predict machine learning models, build convolutional neural networks, and recurrent neural networks, and visualize the results. The benefits of Python include an extensive selection of frameworks and libraries for machine learning and deep learning models. Many libraries are easy to implement and have certain advantages, for example, NumPy and SciPy packages can be used for profound data analysis, the scikit-learn library is widely used for ML model implementation, Keras library uses a graphics processing unit (GPU) in addition to a central processing unit (CPU), also called a central processor, which allows performing calculations more rapid (Patel, 2018).

Machine Learning models use numerical data to train and make predictions, therefore non-numerical features should be transformed, it will be done with the help of LabelEncoder from sklearn.preprocessing package. To transform such features as time and to preserve their cyclical significance CyclicalTransformer from the feature_engine.creation package will be used.

Other machine learning models described will be implemented with the help of the following Python libraries (Table 2).

Table 2. Python libraries used for ML and DL models implementation.

| **Machine Learning model** | **Python package** |
|---|---|
| Logistic regression | sklearn.linear_model.LogisticRegression |
| Decision tree classifier | sklearn.tree.DecisionTreeClassifier |
| Random Forest Classifier | sklearn.ensemble.RandomForestClassifier |
| Naïve Bayes Classifier | sklearn.naive_bayes.GaussianNB |
| K-Neighbors classifier | sklearn.neighbors.KNeighborsClassifier |
| Support Vector Machine Classifier | sklearn.svm.SVC |
| Convolutional Neural Network | keras.models.Model |
| Recurrent Neural Network | keras.models.Sequential |

These Python libraries are open source and are commonly used to speed up the programming and automation process of ML and DL.

# 6. Model application to Biocad`s data

Since most of the data is collected directly from the equipment, there is a problem of loss of the data due to equipment specification, therefore models of predictive maintenance can face the situation of missing data, so that replacement need prediction can become a more difficult task than expected. To resolve the issue, models were trained and compared based on two periods – four-week and two-week before the expiration time.

## 6.1. Biocad`s dataset

The data for the thesis is provided by BIOCAD. 10 datasets from 10 different chromatographs of the same type have been used, this data set concatenated consists of 157 885 rows total.

Dataset consists of three columns with information about chromatograph activity (Table 3).

Table 3. Example of the data provided by BIOCAD

| **date** | **event_class** | **text_message** |
|---|---|---|
| 2016-02-08 08:38:23.000 | log.from.chromatograph | 1 G1316A:DEACN41306 - Thermostat off |
| 2016-02-08 08:39:28.000 | log.from.chromatograph | 1 G1311C:DEAB818811 - Pump initializing |
| 2016-02-08 08:39:29.000 | log.from.chromatograph | 1 G1329B:DEAAC38976 - Get System Ready trigger> |

It is sequential data that present the activity of each chromatograph one after another. In sum, the data frame consists of the information from 10 chromatographs.

The ‘Date’ column represents consecutive DateTime values of equipment logs. The range of the DateTime values is from the 2nd of March of 2015 13:08:18 to the 29th of December of 2021 12:01:22.

‘event_class’ column provides characteristic of ‘text_message’. There are 11 unique values of ‘text_message’ column. They have the following meaning:

1. unit.process.info = Regular Action Notice
2. log.from.chromatograph = Info from chromatographs
3. unit.process.step = Process step notification
4. data.status.invalid = Data not accepted
5. book.service.end = Equipment is free
6. user.event.service =Message from user
7. book.service.start = The user occupied the equipment
8. unit.process.alarm = Emergency message
9. book.fault.end = Equipment is free
10. book.work.end = Equipment is free
11. unit.process.warning = Warning

Term ‘unit.process.info’ is equal to almost 68 % of the dataset, ‘log.from.chromatograph’ – to 31%.

The third column – ‘text_message’ – consists of 2861 unique values. However, it is important to mention that the ‘text_message’ column is presented in two parts – detector and textual information, so data extraction is needed to evaluate an actual number of unique values. ‘Text_message’ column includes important information for the task – among processes description there are also facts of lamps replacement and lamps activity. The column has such text messages as ‘Lamp on’, ‘Lamp off’, so that the operating time of lamps can be calculated and measured, and also the column has information about the maintenance – such text messages as ‘UV-lamp replacement’ can be used to calculate a number of lamps, to create target variable, etc.

There is the data from 10 chromatographs. As can be seen from the graph, the $6^{th}$ and the $7^{th}$ chromatographs contribute a lot to the number of observations. It can lead to misleading results by adding extra weight to the data presented from these two chromatographs, therefore data normalization is needed. Then the number of observations from chromatographs was plotted in Python (Figure 4).

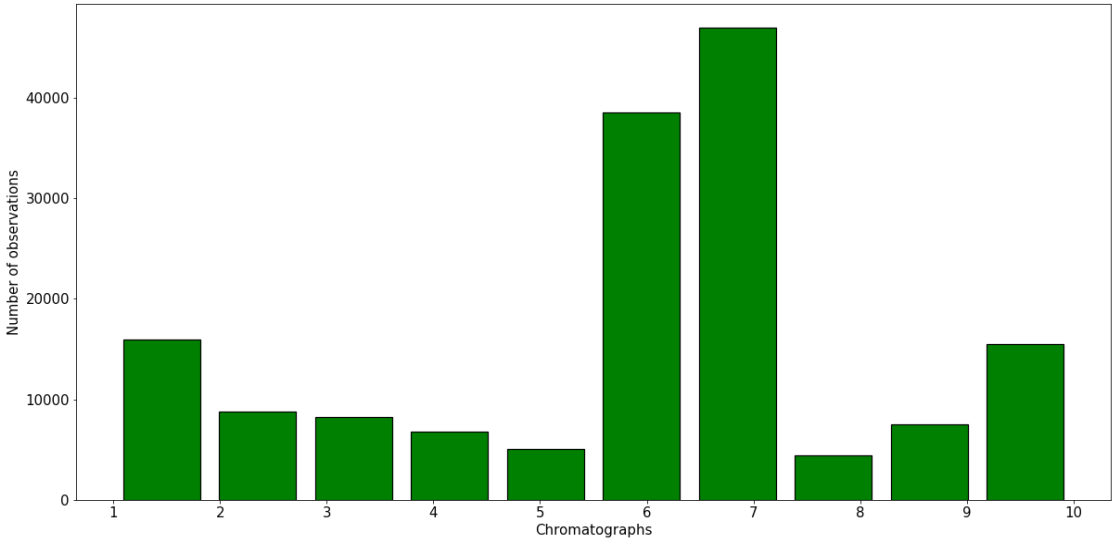


Figure 4. A number of observations from 10 chromatographs.

As can be seen from the graph, the 6th and the 7th chromatographs` logs present the largest number of observations. It can be explained by the active use of these two chromatographs in the analysis. Though their activity logs should be analysed more thoroughly, so that it does not influence models` results due to the presence of outliers.

## 6.2. Data preprocessing

The 'Text_message' column consists of information in two languages: Russian language and English language. Text written in Russian contains important information considering lamp replacement, therefore the text was translated using the 'translate' function. General function for further usage within the company was also implemented using the Python language.

The most important information considering lamp replacement is in the 'text_message' column. If the data is checked using the 'str.contains('replacement of UV')' command, it has the following result (Table 4).

Table 4. Lamps replacement information

| **Index** | **Date** | **event_class** | **text_message** | **Chromatograph** |
|---|---|---|---|---|
| 5178 | 2020-07-23 06:27:04.479 | user.event.service | Completed work: replacement of UV lamp | 1 |
| 5179 | 2020-07-23 06:27:13.000 | book.service.end | Refusal: For an unforeseen reason; Completed work: replacement of UV lamp | 1 |

| | | | | |
|---|---|---|---|---|
| 9672 | 2021-03-29 09:13:59.873 | user.event.service | Completed work: replacement of UV lamp | 1 |
| 9673 | 2021-03-29 09:14:12.000 | book.service.end | Completed work: replacement of UV lamp | 1 |
| 23237 | 2021-06-15 14:20:34.963 | user.event.service | Completed work: replacement of UV lamp | 2 |
| 23238 | 2021-06-15 14:20:37.000 | book.service.end | Refusal: For an unforeseen reason; Completed work: replacement of UV lamp | 2 |

It is obvious, that rows are duplicated because of additional information from the second event_class. As BIOCAD staff explained, string ‘Refusal: For an unforeseen reason. Completed work: replacement of UV lamp’ basically means the same as ‘Completed work: replacement of UV lamp’. Therefore, information considering lamps replacement from book.service.end will be removed, because user.event.service consists of the information from the user of the equipment which is more relevant in this case. After the removal of duplicated rows, we have less than 1% of lamps replacement cases in the dataset. This is highly imbalanced data.

Through the initial data analysis, the first challenges were discovered. Firstly, information about lamps replacement is available only for 2020-2021, while a range of date values is from 2016 till 2021. Secondly, for each chromatograph operating hours value is available only at one point of the time. Therefore, there is a difficulty to calculate accurately lamp working hours.

For several chromatographs, calculations had to be done manually to have accurate results. Moreover, chromatographs have non-uniform activity: high activity periods are succeeded by the standstill periods. Company representatives assured us that this is a usual practice because chromatographs can operate at night or at the weekends. As already mentioned, the data represents an imbalanced classification problem. Initial data has less than 0,02 % of values labeled as 1 (Positive Class (Failure)).

‘text_message’ column was used to conduct feature creation. First, the column was split into two other columns – one with the detector name and the other with text information. Since machine learning models use numerical data to train and predict classes, textual information must be transformed into a machine-readable format. There are several methods for data preprocessing, the two most widely used are One Hot encoder, Label Encoder, and Ordinal Encoder. One hot encoder creates a new column from each separate feature, while the Label encoder presents the data in a categorical way – for example, by naming the data as 3,2,1. The disadvantage of Label Encoder is

that features can be not ordinal, so for the model, it can be a misleading way of presentation, though it uses less memory, so it is more convenient and faster. As for One Hot encoder, its disadvantage is high cardinality, and the researcher would need to tackle the problem of curse dimensionality. Ordinal Encoder assigns an integer value for each unique category, so that common relations between variables are preserved. The data consists of sequential data that has relations between variables, so an Ordinal Encoder was used to transform textual information into a machine-readable format. Then DateTime values were extracted from the date column – the data now has new columns such as ‘Date’, ‘Day of week’, ‘Year’, ‘Month’, ‘Day’, ‘Hour’, ‘Minute’.

Column ‘lamp_number’ was added as values between indexes which contained information considering lamps replacement information and till the end of the next chromatograph, due to lack of data it is impossible to calculate accurately operating time of lamps before the period 2020-2021, so it was considered that new chromatograph starts operating with the new lamp. In sum, we have information considering 34 lamps. The data was grouped by these lamps and calculated the operating time of lamps as a cumulative sum between rows presenting the time value of the start and the end of the work of lamps.

Company representatives stated that the estimated operation time of the lamp before the replacements is about 3000 hours, however, it is a general recommendation to use the lamp not more than 3000 hours. Quite often lamps are operating for a wider period, however, the analysis cannot be conducted successfully in such circumstances and the whole research process will be spoiled. A histogram of the operating time of lamps looks as following (Figure 5).

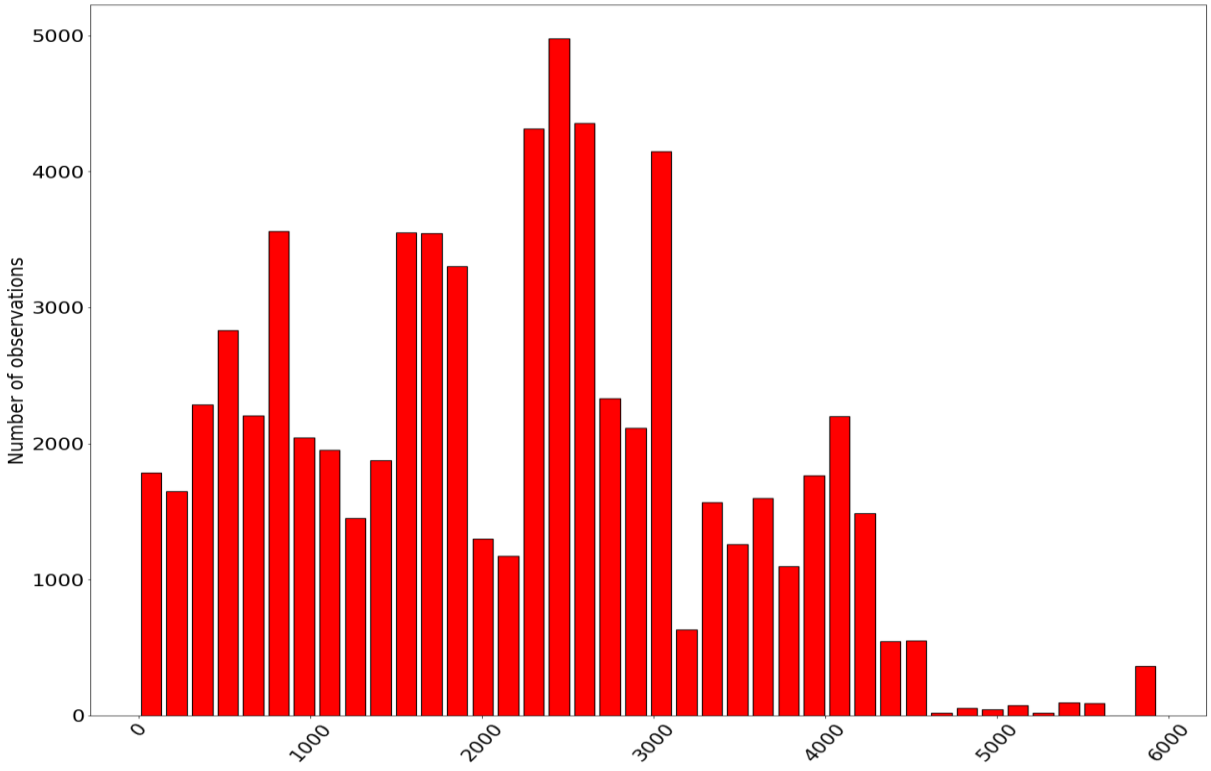


Figure 5. Operating time of lamps.

As can be seen from the graph, the majority of the lamps operate till the benchmark of 3000 hours, though there is a substantial part of lamps that continues operating till 4500 hours, and a small number that is not replaced till the 6000 hours of operating time. As was already mentioned, work of lamps continued after the expiration threshold can worsen analysis results and lead to loss of money and resources, therefore predictive maintenance approach is needed.

Since the data can include meaningful information, feature creation was conducted. There might be some hidden patterns such as hours of the day, days of the week, months, and seasons in the date column. To preserve the cyclical significance of time, CyclicalTransformer was used. It converts original DateTime values to cosine and sine values so that every moment within a period can be assigned with unique values. The disadvantage of this method is that each feature is now represented by two new columns – which will mathematically assign more weight to it from the algorithm perspective and several algorithms cannot process two features simultaneously (Pierre-Louis Bescond, 2020).

The next step of data preprocessing was to get information about detectors and to get cleaned textual data. Initially, these columns represented the following information (Table 5).

Table 5. Information about detectors in the initial data provided by BIOCAD

| **date** | **event_column** | **text_message** |
|---|---|---|
| 2016-02-08 08:39:28 | log.from.chromatograph | 1 G1311C:DEAB818811 - Pump initializing |
| 2016-02-08 08:39:29 | log.from.chromatograph | 1 G1329B:DEAAC38976 - Get System Ready trigger> |

The text was split by the ‘ - ’ symbol with the help of the split() function and the second element (text) was added to the new column ‘text_message_new’. Several values in the ‘text_message’ column also had a ‘>’ symbol which disturbed the function; therefore, it was removed using replace () function. The result looks as following (Table 6).

Table 6. Text extraction result

| **text_message** | **text_message_new** |
|---|---|
| 1 G1311B:DEAE415578 - Changed Actual Solvent A> | Changed Actual Solvent A |
| G1315D:DEAAX10019 - Lamp on | Lamp on |
| 1 G7129A:DEAEQ21405 - Switch valve to Bypass. | Switch valve to Bypass |

The detector label was added using the split() function by ‘:’ and ‘ – ’ symbols. Detector number can also identify lamp replacement – when a certain type of detector changes, the lamps were replaced, therefore this information can be relevant for models training. The result looks as following (Table 7).

Table 7. Detector label extraction result

| **Text_message** | **detector** |
|---|---|
| G7121B:DEAEJ00900 – Run | DEAEJ00900 |
| G7116A:DEAEM02262 – Valve present | DEAEM02262 |

Functions that can be used further within the company for data preprocessing were also implemented using Python.

Since there is no information about measurements or other continuous values, text messages considering any errors, failures, or pressure conditions were extracted and transformed into numeric values using an Ordinal Encoder.

Then a correlation analysis was conducted using the heatmap function from the seaborn package. The correlation matrix looks as following (Figure 6).

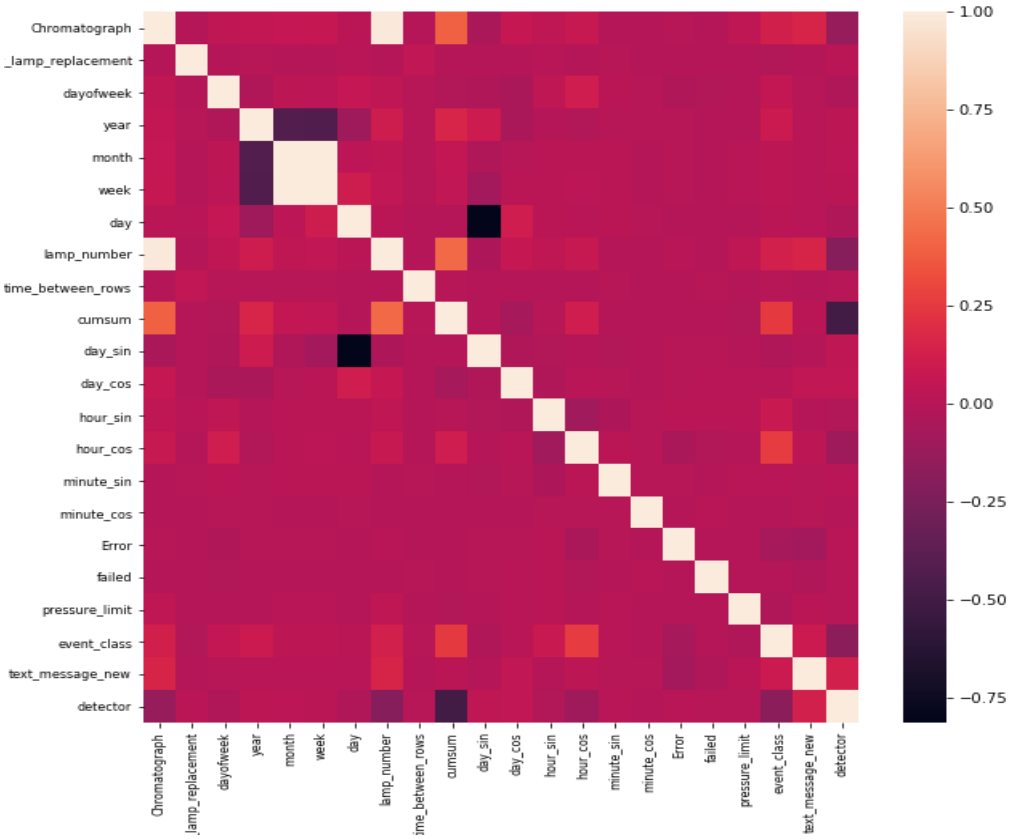


Figure 6. Correlation matrix.

Target variable ‘new_lamp_replacement’ does not have any meaningful correlations with features, while there are strong correlations between explanatory variables, so features were selected using the feature extraction method.

Then two data sets were created, and 0 in the target variable (‘new_lamp_replacement’) was replaced with 1 (lamp replacement) defining the window length by the condition of the operating time of lamps – whether it is within four weeks or two weeks before expiration time (3000 hours).

More precisely, while, in the first case, if the lamp`s operating time is within 2328-3000 hours period (4-week interval), it is considered that the replacement is needed, in the second, if the lamp`s operating time is within 2664-3000 hours period (2-week interval), it is considered that the replacement is needed. The first modification raised the percentage of lamps replacements identifications up to 9% from 0.02%, the second one – to 5% from 0.02%. All lamps with operating time greater than 3000 hours are considered as not needing the replacement, since they can represent outliers, or the calculations can be not accurate due to not systematic data obtained from the chromatographs.

Several machine learning models are highly affected by features weights, so the features were scaled before applying them. I used StandardScaler() function to standardize features onto a unit scale with the mean equal to 0 and variance equal to 1, the approach helps to avoid the problem of losing the outliers.

Then feature extraction was conducted using the Extra Trees Classifier. This model is also known as Extremely Randomized Trees. The result looks as following (Figure 7).

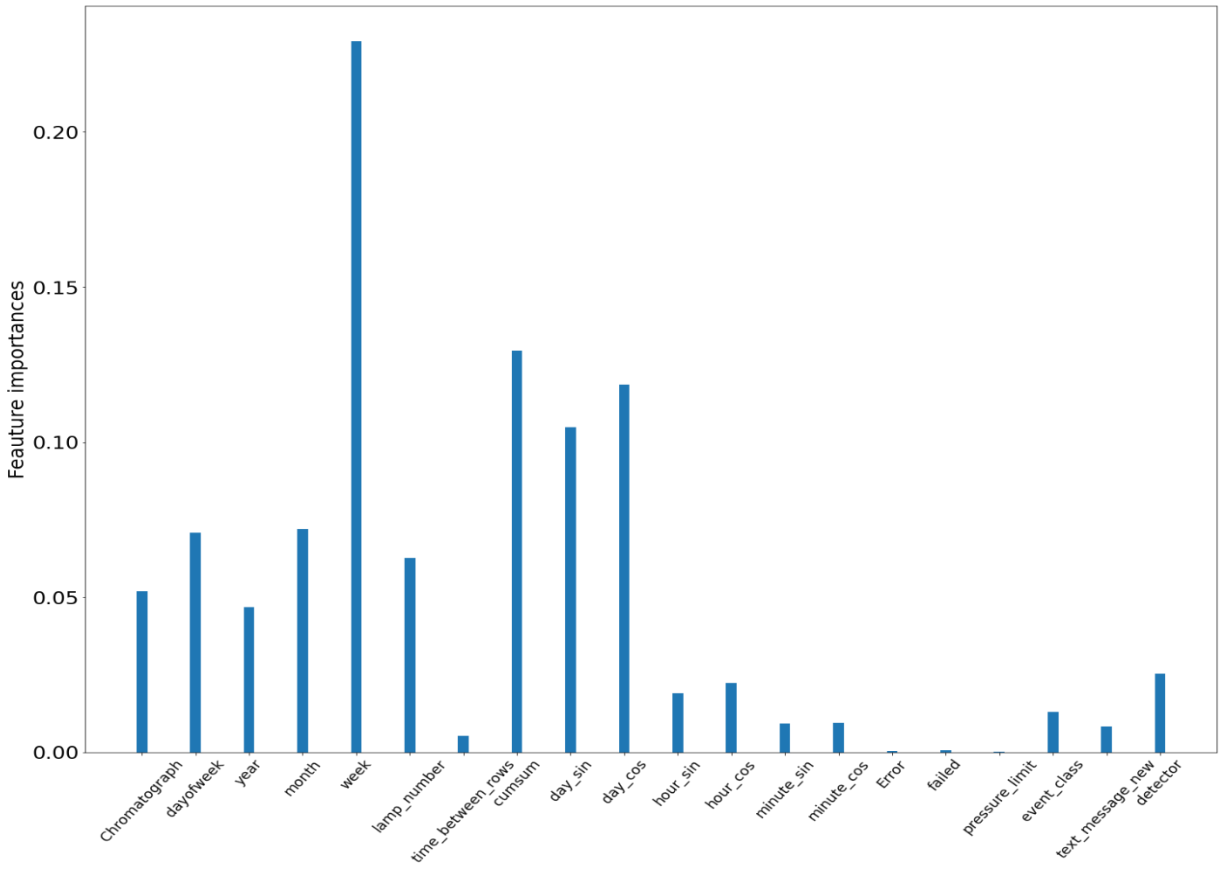


Figure 7. Feature importance by Extra Trees Classifier (4-week interval).

As can be seen from the graph, several features contribute the most to the explanation of the variance of the target variable – lamp replacement. Among them are the number of the chromatograph, year, month, week, lamp number, the day represented by two columns, and operating times of lamps. In sum, they explain about 81% of the lamp replacement value`s variance.

Different data preprocessing methods were used, so an analysis using DecisionTreeClassifier was conducted. The Decision Tree Classifier approach is based on building the decision tree, while the Extra Trees Classifier selects a random value for the split for each feature in the data just like the Random Forest model. Usually, Extra Trees Classifier reduces the risk of overfitting while Decision Tree Classifier, on the contrary, shows a high variance of the predictions (Naman Bhandari. 2018). The results of the feature importance analysis look as following (Figure 8).

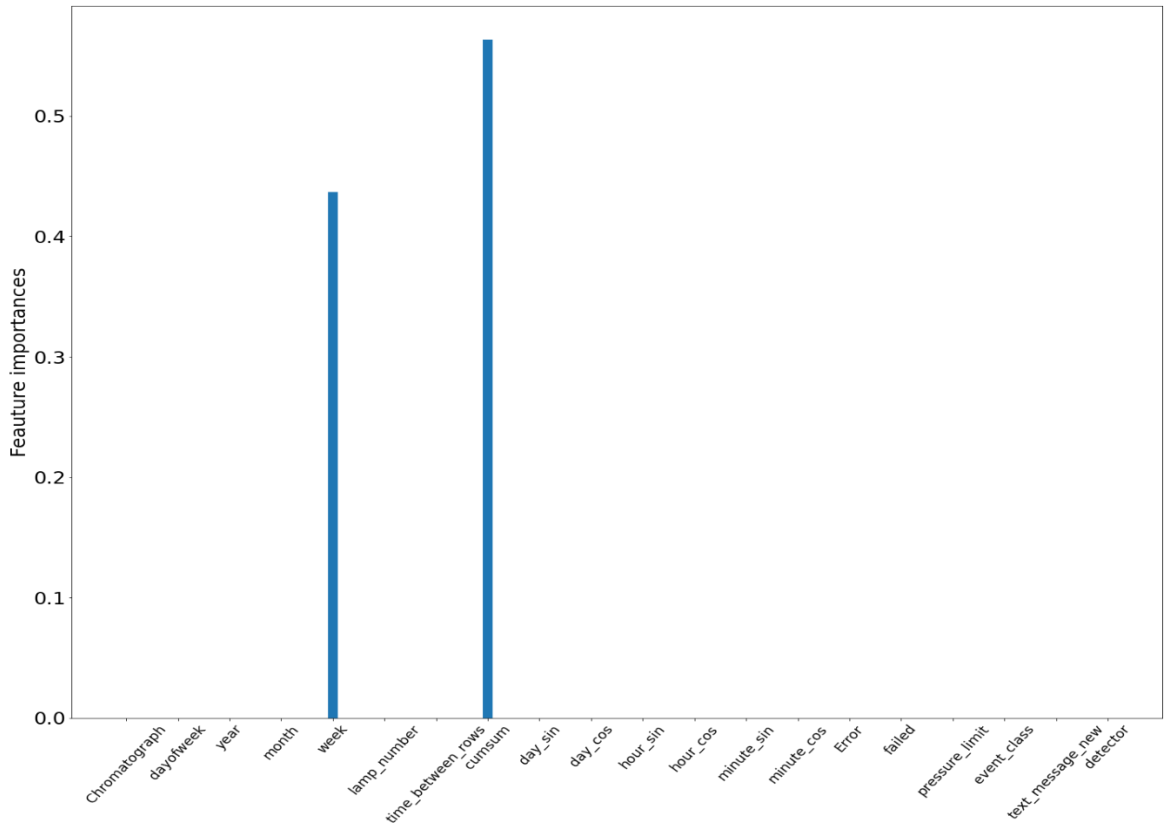


Figure 8. Feature importance by Decision Tree Classifier (4-week interval).

As can be seen from the graph, results differ, having a 100% of variance of the target variable explained by two features – the operating time of lamps and a week. From figure 8 it is obvious that these features are significant too.

Since Extra Trees Classifier has a lower variance in comparison to Decision Tree classifier, because the former reduces the risk of overfitting by introducing randomness using random subsets of features, features were selected based on the ExtraTreesClassifier result.

After feature selection is done with the help of ExtraTreesClassifier so that 81% of target variable variance is described by selected features. The data looks as following (Table 8).

Table 8. Data preprocessing result

| **Chromatograph** | **year** | **month** | **week** | **lamp_number** | **cumsum** |
|---|---|---|---|---|---|
| 0 | 0.166667 | 0.090909 | 0.096154 | 0 | 0.605291 |
| 0 | 0.166667 | 0.090909 | 0.096154 | 0 | 0.605291 |
| 0 | 0.166667 | 0.090909 | 0.096154 | 0 | 0.605291 |

The data set contains 6 columns and 83461 rows. All features are normalized, so that they can be used for ML and DL models training.

## 6.3. Results of data classification using ML models

Since there are two boundaries for the period when the lamp is considered as needed the replacement, models were trained on two different data sets – based on a four-week interval and a two-week interval – to compare which models present significant results within this confidence interval.

The data was split into three parts – train, validation, and test data. A validation set is used for tuning the parameters of the model, a test set is used to evaluate the performance of the model with selected parameters. In this sense, validation data occurs as part of model training.

### 6.3.1. Prediction based on four-week interval

Several machine learning models were used to classify the data and have conducted hyperparameter selection, which resulted in higher values of recall and precision. For example, the change of distance metric from Minkowski to Hamming in the KNN-classifier raised the precision score from 0.89 to 0.99 and raised the recall score from 0.93 to 0.99. Logistic regression was tuned by changing the threshold to 0.2, which raised the precision score from 0 to 0.42, and the recall score from 0 to 0.38, so that F-score increased from 0.02 to 0.4. The model was validated through the stratified 10-fold cross-validation. The mean results of four metrics for each model look as following (Table 9).

Table 9. Cross-validation results based on the four-week interval (ML)

| **ML model** | **Mean accuracy** | **Mean precision** | **Mean recall** | **Mean F-score** |
|---|---|---|---|---|
| Logistic Regression | 0.95 | 0.41 | 0.38 | 0.4 |
| Support Vector Machines | 0.95 | 0.01 | 1 | 0.01 |
| Decision Trees | 1 | 1 | 1 | 1 |
| Random Forest | 1 | 1 | 1 | 1 |
| Naive Bayes | 1 | 0.77 | 0.78 | 0.77 |
| K-Nearest Neighbor | 1 | 0.99 | 0.99 | 0.99 |

As can be seen from the table, the Decision Tree classifier, Random Forest classifier, and K-Nearest Neighbor classifier provided the best results in terms of four metrics.

The models' ability to predict lamp replacement necessity on the test data was checked. The results look as following (Table 10).

Table 10. Classification results based on the four-week interval (ML)

| **ML model** | **Accuracy** | **Precision** | **Recall** | **F-score** |
|---|---|---|---|---|
| Logistic Regression | 0.93 | 0.01 | 0.78 | 0.01 |
| Support Vector Machines | 0.95 | 0.01 | 1 | 0.01 |

| | | | | |
|---|---|---|---|---|
| Decision Trees | 1 | 1 | 1 | 1 |
| Random Forest | 1 | 1 | 1 | 1 |
| Naive Bayes | 0.98 | 0.77 | 0.82 | 0.79 |
| K-Nearest Neighbor | 0.98 | 0.97 | 0.98 | 0.97 |

As expected, the test data results are worse than predictions obtained through cross-validation (Table 9), however, the models' prediction ability did not decrease significantly, some of the algorithms repeatedly showed the ability to make accurate predictions, among them are Decision Tree Classifier, Random Forest Classifier, and KNN algorithm. Results were also visualized using the matplotlib library (Figure 9).

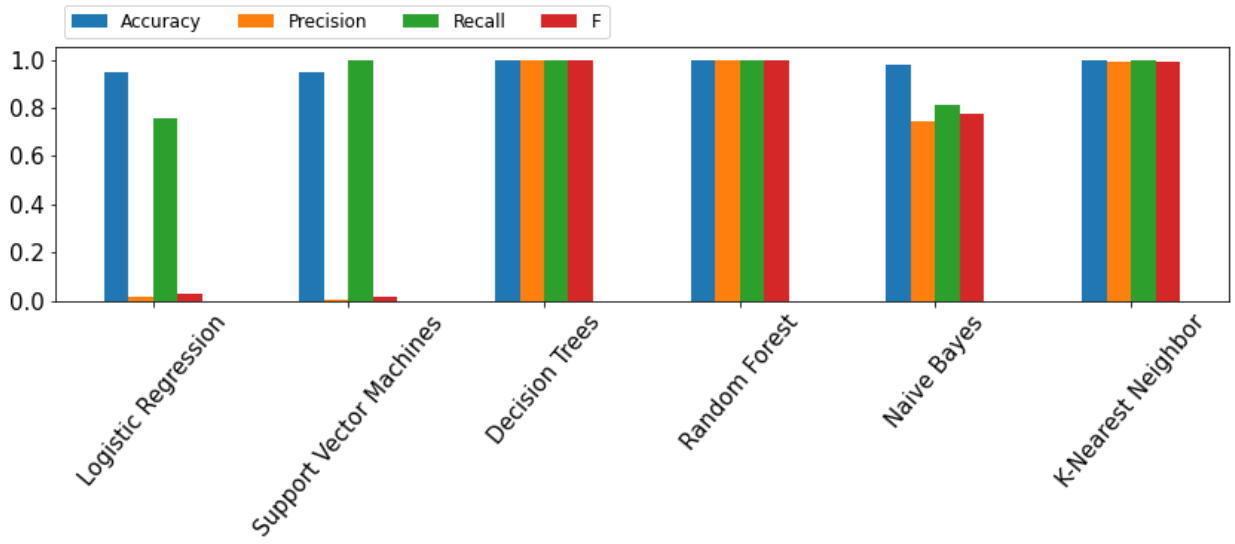


Figure 9. Classification results (visualization) based on the four-week interval (ML).

Since three models made the predictions accurately, the reasoning behind the calculations of the algorithm can be discussed. Decision Tree showed high scores for all selected metrics, and decision rules that helped to achieve such a promising result were extracted (Figure 10).

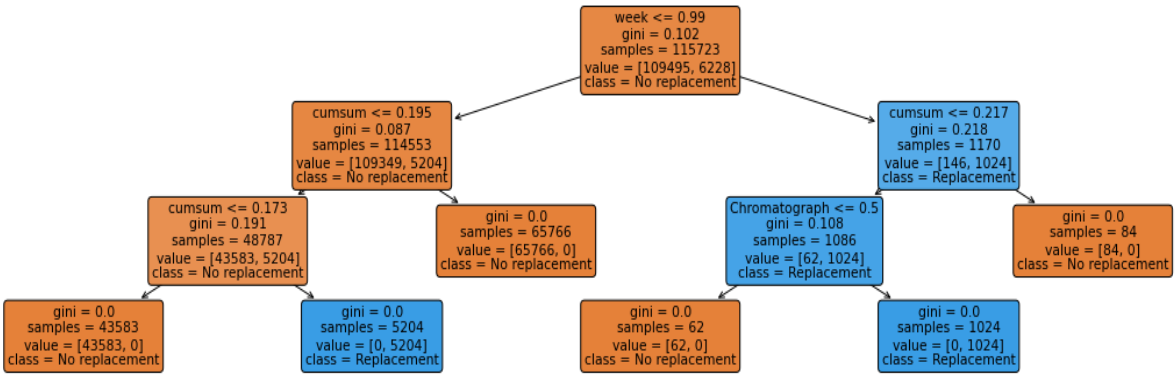


Figure 10. Decision rules based on four-week interval.

As can be seen from the figure, the tree is not deep, and the rules are quite understandable. Such features as a week, operating time of the lamps (‘cumsum’), and the serial number of the chromatograph provided an easy approach for effective calculations. Gini criterion was selected to make predictions.

The rules can also be presented as a textual list:

```
if week <= 0.9903846085071564:
    if cumsum <= 0.19475824385881424:
      if cumsum <= 0.17295384407043457:
        return [[43583. 0.]]
      else:  # if cumsum > 0.17295384407043457
        return [[0. 5204.]]
    else:  # if cumsum > 0.19475824385881424
      return [[65766. 0.]]
  else:  # if week > 0.9903846085071564
    if cumsum <= 0.21698933839797974:
      if Chromatograph <= 0.5000000149011612:
        return [[62.  0.]]
      else:  # if Chromatograph > 0.5000000149011612
        return [[0. 1024.]]
    else:  # if cumsum > 0.21698933839797974
      return [[84.  0.]]
```

The return statement shows how many samples were classified as the first or the second class. For example, in the last row, the model returns 84 samples of lamps that do not need replacement and 0 objects of the lamps that need replacement. All features were normalized before the models’ training, so they are presented in the [0,1] range. The first rule of the Decision Tree Classifier is based on the week number. Only fifty-third week satisfies the following condition: ‘week > 0.99’, since 1 in normalized data represents 53 in not normalized data. 53 is also the largest number of

the week among the observations. It means that there is a dependence of lamp replacement needs on the sequence number of the week, which also signifies the influence of the time-related data. Within this if-statement (week > 0.99), the 'Chromatograph' number is also crucial, 'Chromatograph > 0.5' leads to the accurate prediction of 1024 samples of replacement need, it can be explained by the presence of a large number of observations from the $6^{th}$ and the $7^{th}$ chromatographs in the data, since their chromatograph numbers are presented in normalized data as 0.55 and 0.66 respectively. In the if-statement 'week <= 0.99' operating time of the lamps was the key feature that influenced the decision-making process. The operating time of not normalized data corresponding to the condition 'cumsum > 0.17' in normalized data reflects the selected four-week interval for lamp replacement consideration. The presence of outliers can result in uncertainty of the results, which is reflected by the if-statement 'cumsum > 0.19', which results in the prediction of 65766 samples of lamps that do not need the replacement.

Another model that showed good results is the Random Forest. Since it is a combination of Decision Trees, it is not convenient to visualize it. However, decision rules and general information about the classifier can be extracted. Random Forest Classifier consists of 100 estimators. Almost all individual trees in the Random Forest classifier are deep (30 splits or more), they are much more complicated than the Decision Tree classifier, for example, the first estimator bases its predictions on 61 rules.

KNN-classifier can be visualized if the features are presented in two-dimensional space, however, in terms of multi-dimensional problems it's visualization and analysis of the calculations are quite limited. As was already mentioned, Hamming distance increased the precision and recall results, number of K can also be extracted from the model, it is equal to 5.

### 6.3.2. Prediction based on a two-week interval

To get predictions for the narrower period (based on a two-week interval), the data was pre-processed using the same steps. Extra Tree Decision classifier showed a similar trend with slight changes in values (Figure 11).

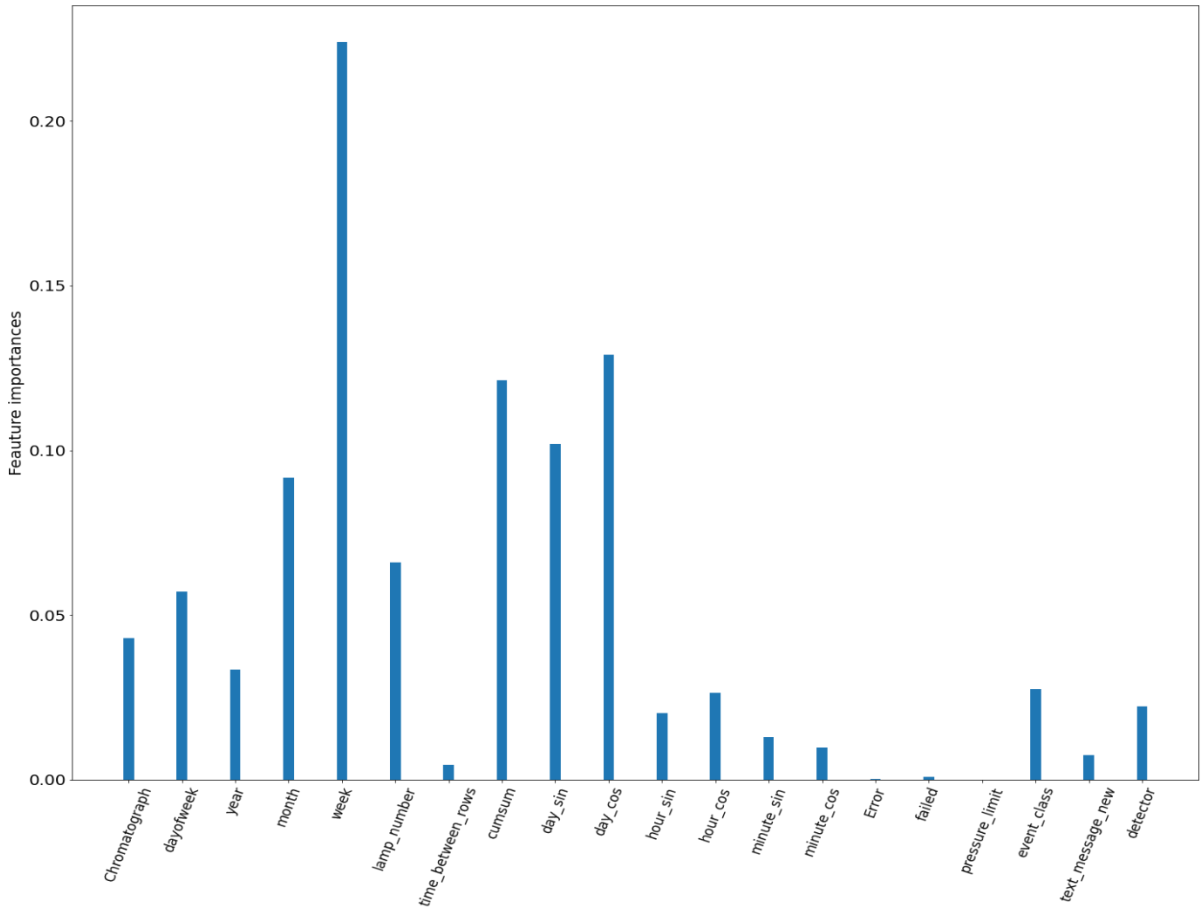


Figure 11. Feature importance by Extra Trees Classifier (2-week interval).

As can be seen from the graph, several features contribute the most to the explanation of the variance of the target variable. Among them are a number of the chromatograph, day of week, month, week, lamp number, the day represented by two columns, and operating times of lamps. In sum, they explain about 78% of the lamp replacement value`s variance.

Selected models were also assessed with the help of stratified 10-fold cross-validation. Change of hyperparameters in classifiers helped to increase the result, Hamming distance was again the best in terms of increasing precision and recall values for KNN-classifier. Change of the threshold of logistic regression also increased several metrics – precision, recall, and f-score. The result looks as following (Table 11).

Table 11. Cross-validation results based on a two-week interval (ML)

| ML model | Mean accuracy | Mean precision | Mean recall | Mean F-score |
|---|---|---|---|---|
| Logistic Regression | 0.98 | 0.02 | 0.9 | 0.05 |
| Support Vector Machines | 0.98 | 0.02 | 0.9 | 0.04 |
| Decision Trees | 1 | 1 | 1 | 1 |
| Random Forest | 1 | 1 | 1 | 1 |
| Naive Bayes | 0.96 | 0.27 | 0.99 | 0.43 |
| K-Nearest Neighbor | 1 | 1 | 1 | 1 |

The model's ability with selected parameters to predict lamp replacement on the test data was checked. The result looks as following (Table 12).

Table 12. Classification results based on the two-week interval (ML)

| ML model | Accuracy | Precision | Recall | F-score |
|---|---|---|---|---|
| Logistic Regression | 0.99 | 0.03 | 1 | 0.05 |
| Support Vector Machines | 0.98 | 0.02 | 1 | 0.04 |
| Trees | 1 | 1 | 1 | 1 |
| Random Forest | 1 | 1 | 1 | 1 |
| Naive Bayes | 0.96 | 0.97 | 0.26 | 0.41 |
| K-Nearest Neighbor | 1 | 0.99 | 1 | 0.99 |

Results based on a two-week interval are almost the same, there is a general trend of good prediction ability among such models as Decision Tree, Random Forest, K-Nearest Neighbor classifiers, and not a precise prediction of such models as logistic regression and Support Vector Machine. All models except one showed almost identical results in terms of four metrics, while Naïve Bayes Classifier recall results decreased significantly. The metrics were also visualized using the matplotlib library (Figure 12).

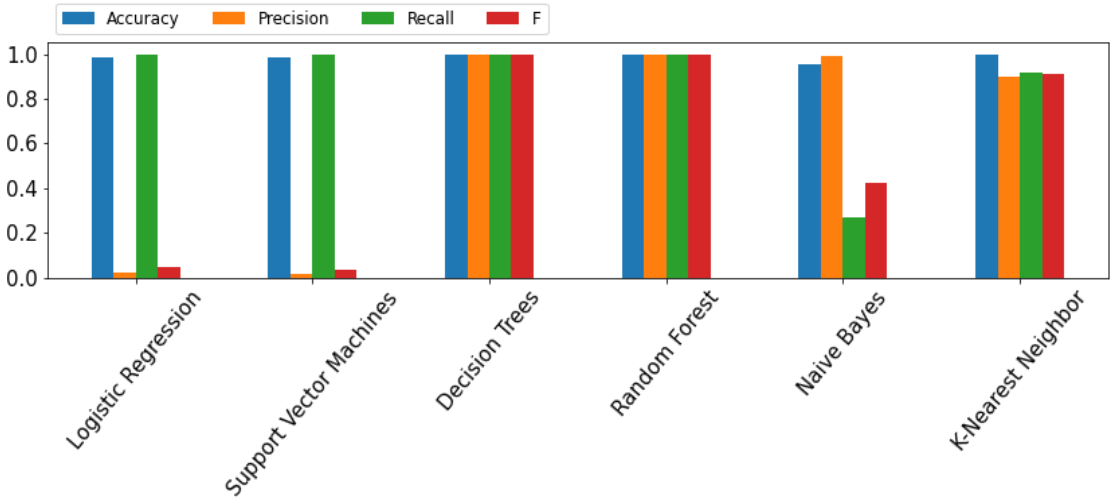


Figure 12. Classification results (visualization) based on the two-week interval (ML).

As can be seen from the figure 12, three models show the best results in terms of four metrics.

Identically  Decision Tree Rules were extracted (Figure 13).

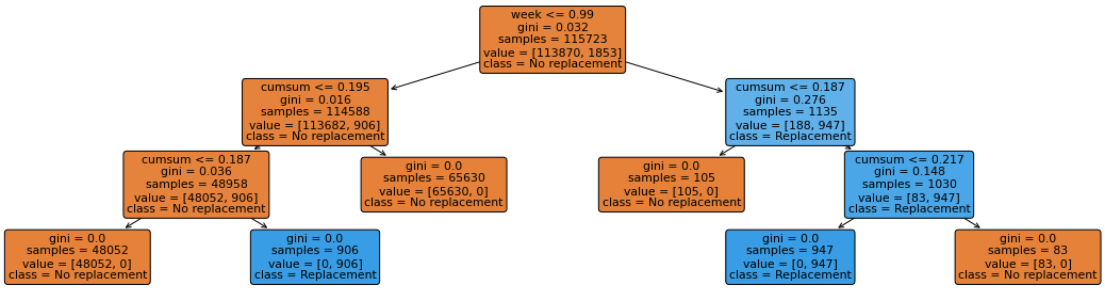


Figure 13. Decision rules based on a two-week interval.

The rules can also be presented in the textual format. The Decision Tree Classifier used the following rules:

```
if week <= 0.9903846085071564:
    if cumsum <= 0.19475824385881424:
      if cumsum <= 0.18697745352983475:
        return [[48052. 0.]]
      else:  # if cumsum > 0.18697745352983475
        return [[0. 906.]]
    else:  # if cumsum > 0.19475824385881424
      return [[65630.     0.]]
  else:  # if week > 0.9903846085071564
    if cumsum <= 0.1869799792766571:
      return [[105.   0.]]
    else:  # if cumsum > 0.1869799792766571
      if cumsum <= 0.2169889733195305:
        return [[0. 947.]]
```

```
else:  # if cumsum > 0.2169889733195305
  return [[83.  0.]]
```

The algorithms are quite similar, usage of the two features – operating time of lamps(‘cumsum’) and week – were the most important in terms of making predictions by Decision Tree Classifier. However, while the ‘Chromatograph’ feature is not presented here, it was significant for predictions based on a 4-week interval. All features were normalized before the models’ training, so they are presented in the [0,1] range. The decision rules are based on the fact whether the week number is more than 0.99 or not, this situation is discussed in the ‘Prediction based on four-week interval’ chapter (6.3.1.). Other decision rules are based on the operating time calculations, the condition ‘cumsum > 0.18’ in normalized data reflects the selected two-week interval for lamp replacement consideration. The presence of outliers influences the decision-making process, which is reflected by the if-statement ‘cumsum > 0.19’, which results in the prediction of 65630 samples of lamps that do not need the replacement.

As for Random Forest Classifier, the algorithm based on the two-week interval is more complex, the mean value of rules among estimators is 80, and in general, trees are deeper. KNN-classifier hyperparameters remained the same.

## 6.4. Results of data classification using DL models

For further analysis, Neural Network and Long-Short Term memory models were also used.

### 6.4.1. Prediction based on four-week interval

A function to build a CNN model based on provided data was implemented. It has 3 layers and the SoftMax function is used as the last activation function. This algorithm uses batch normalization, average pooling, and the ReLu function for calculations. Batch Normalization is usually used to make the NN model prognosis more stable by re-centering and re-scaling layers` inputs. Average pooling takes a tensor and computes the average value among all the values (Ciresan et al., 2011).

Rectified Linear units (ReLu) function is a widely used method to account for non-linearities. The function returns 0 for any negative input, and it returns a value for any positive input (Liang, 2021). The formula looks like the following one:

$$f(x) = \max(0, X) \quad (12)$$

ReLU function is represented by the following graph (Figure 14):

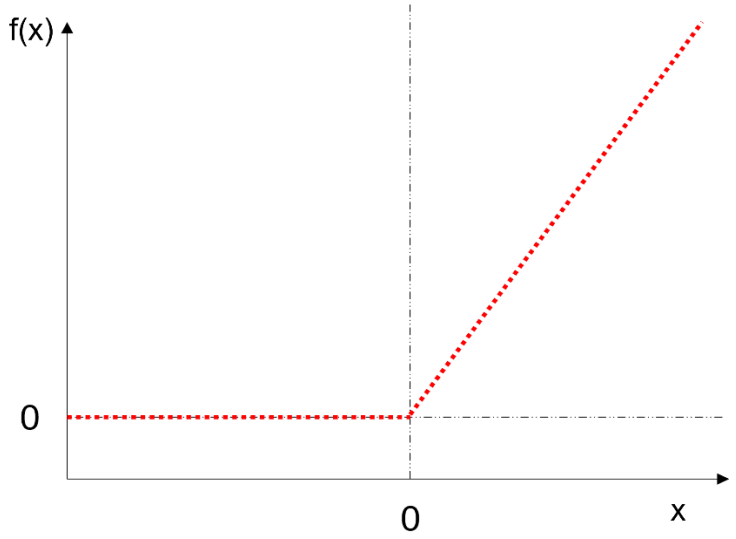


Figure 14. ReLU function. Image is adapted from (Szandała, 2018).

The batch size of the model is equal to 8, and the number of epochs is equal to 50, which is based on the hyperparameter selection conducted. For the model`s results estimation accuracy, precision and recall were used.

The model was validated with the help of stratified 10-fold validation. Since Keras library does not have an in-built function for f-score estimation, it was added manually to the function. The model predictions showed the following mean of metrics (Table 13).

Table 13. Cross-validation results based on the four-week interval (NN)

| **Mean accuracy** | **Mean precision** | **Mean recall** | **Mean F-score** |
|---|---|---|---|
| 0.99 | 0.01 | 0.01 | 0.01 |

The results of the model are not significant, meaning that the algorithm can not make predictions accurately. This can be related to the presence of an overfitting problem or misspecification of the model.

Then the model was used to make predictions based on the test data. The results look as following (Figure 15):

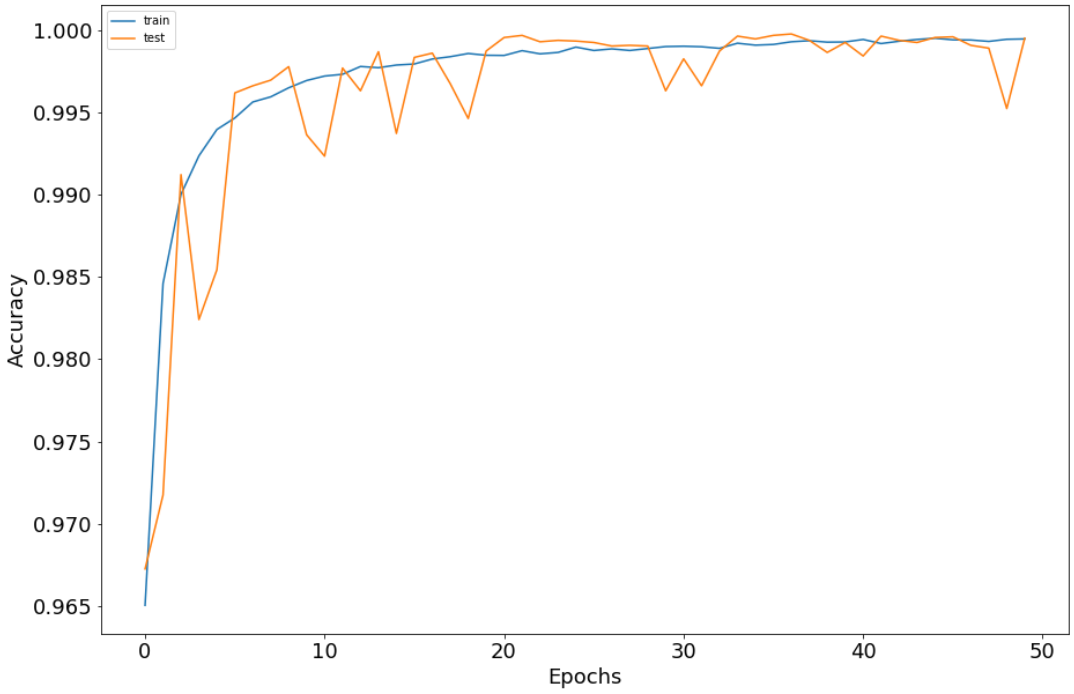


Figure 15. CNN model prediction accuracy based on a four-week interval.

As can be seen from the figure, the accuracy of the model is quite high for both train and test values. However, as was already discussed, high accuracy score in imbalanced data can be a misleading metric, so other metrics should also be observed. The precision score results look as following (Figure 16):

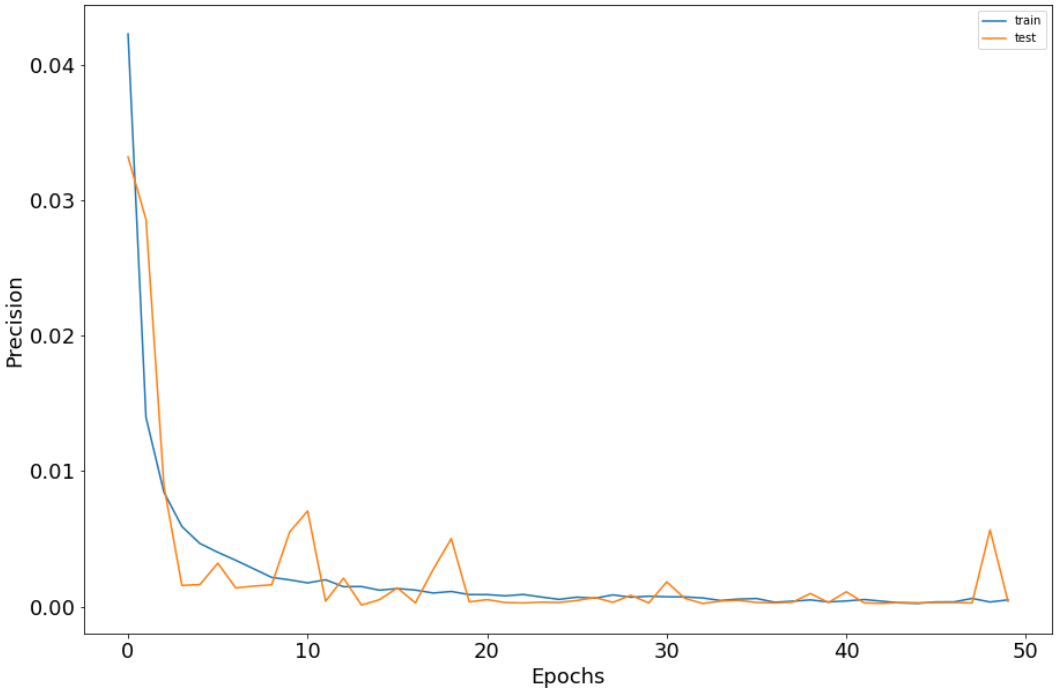


Figure 16. Precision values based on the four-week interval (CNN).

The recall score results look as following (Figure 17):

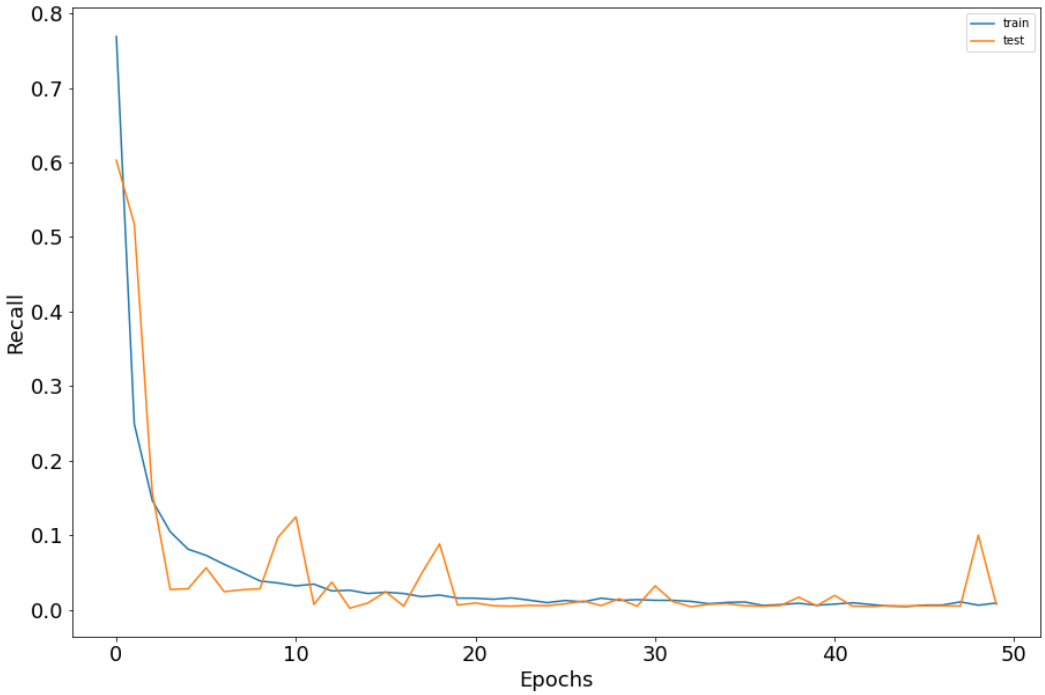


Figure 17. Recall values based on the four-week interval (CNN).

Results for precision and recall values are almost 0. Precision and recall curves followed the same trend but with a different score of the metric.

For LSTM-model another function was implemented in Python, the Sequential () function was used. As the last activation function SoftMax was chosen. To evaluate the loss of the model, the same metrics were used. Model`s epochs are equal to 100 and batch size is equal to 72, which is based on hyperparameter selection conducted.

LSTM-model was also validated with stratified 10-fold cross-validation. The result looks as following (Table 14).

Table 14. Cross-validation results based on the four-week interval (LSTM)

| **Mean accuracy** | **Mean precision** | **Mean recall** | **Mean F-score** |
|---|---|---|---|
| 0.99 | 0.97 | 0.9 | 0.93 |

Cross-validation scores are high, meaning that the model can predict the need for the lamp replacement accurately.

Then the model was used to make predictions based on the test data. The accuracy score results look as following (Figure 18).

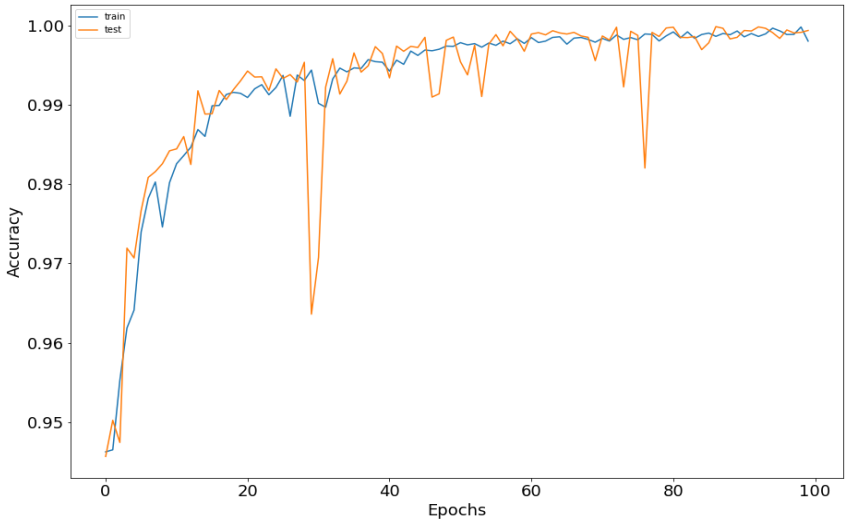


Figure 18. LSTM-model prediction accuracy based on a four-week interval.

As can be seen from the figure 18, the accuracy score is increasing while model training, moreover, test set predictions are getting more stable while epochs increase.

The results of precision and recall values for the Long-Short Term Memory model were observed. Precision score results look as following (Figure 19).

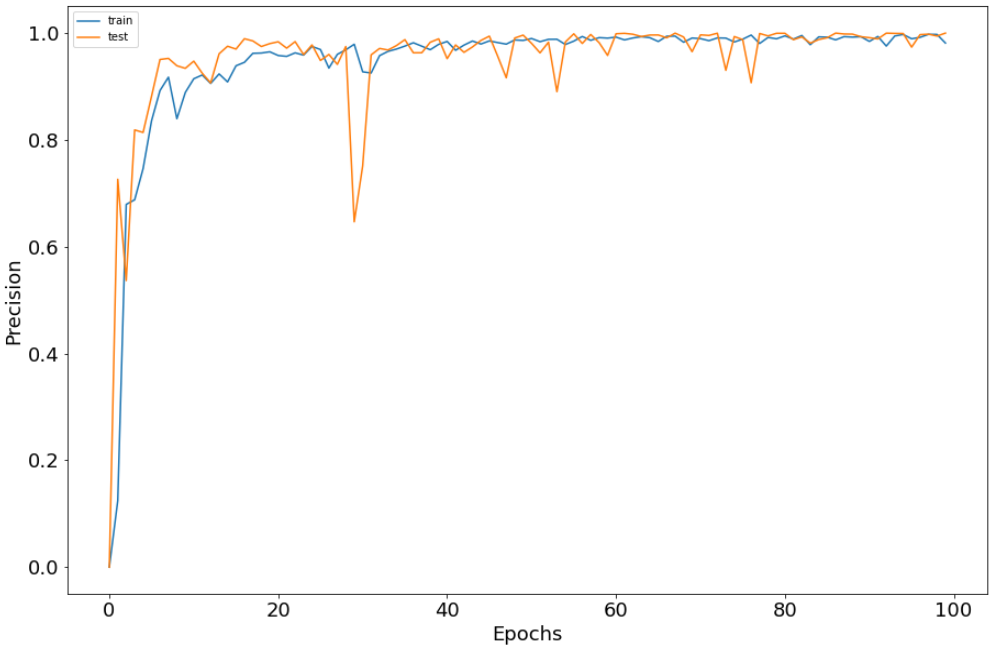


Figure 19. Precision values based on the four-week interval (LSTM).

Recall score results look as following (Figure 20).

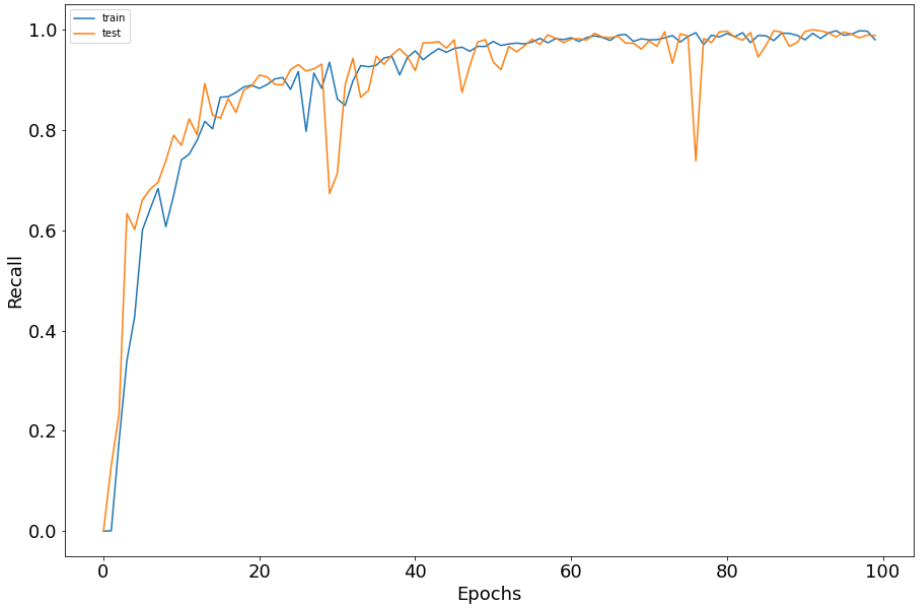


Figure 20. Recall values based on the four-week interval (LSTM).

Results are almost 1 for both recall and precision while epochs increase. Therefore, predictions based on a four-week interval by the LSTM model are much better than by the Convolutional Neural Network model.

### 6.4.2. Prediction based on a two-week interval

The same models were run based on the narrower period. Those rows that are within two weeks before expiration time (3000 hours) were marked as '1' (an indicator of lamp replacement). Lamps with an operating time of more than 3000 hours are considered as not needed the replacement, because they can represent the outliers, or calculations can be not accurate due to not systematic data obtained from the chromatographs. The results of cross-validation look as following (Table 15).

Table 15. Cross-validation results based on the two-week interval (CNN)

| **Mean accuracy** | **Mean precision** | **Mean recall** | **Mean F-score** |
|---|---|---|---|
| 0.99 | 0.01 | 0.01 | 0.01 |

The table shows results identical to the cross-validation results based on a four-week interval by CNN. Then the model was used to make predictions based on the test data. The accuracy score results look as following (Figure 21):

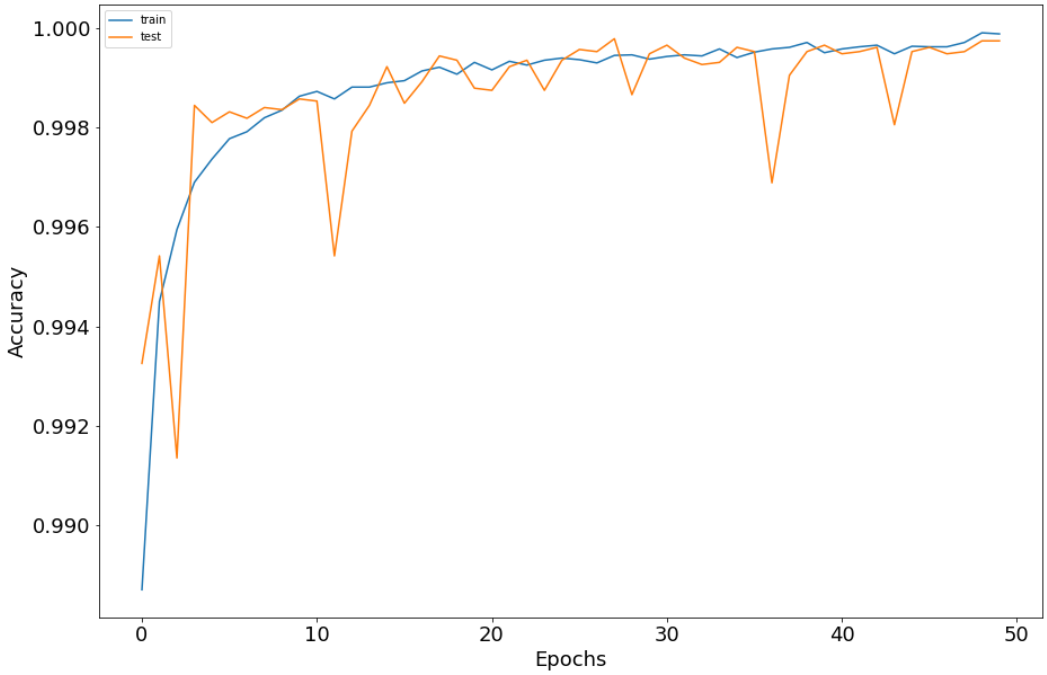


Figure 21 CNN-model prediction accuracy based on a two-week interval.

Results for the accuracy are quite high, though it is a misleading result. Then d the results of the Neural Network model`s precision and recall values based on a two-week interval were observed. Precision score results look as following (Figure 22).

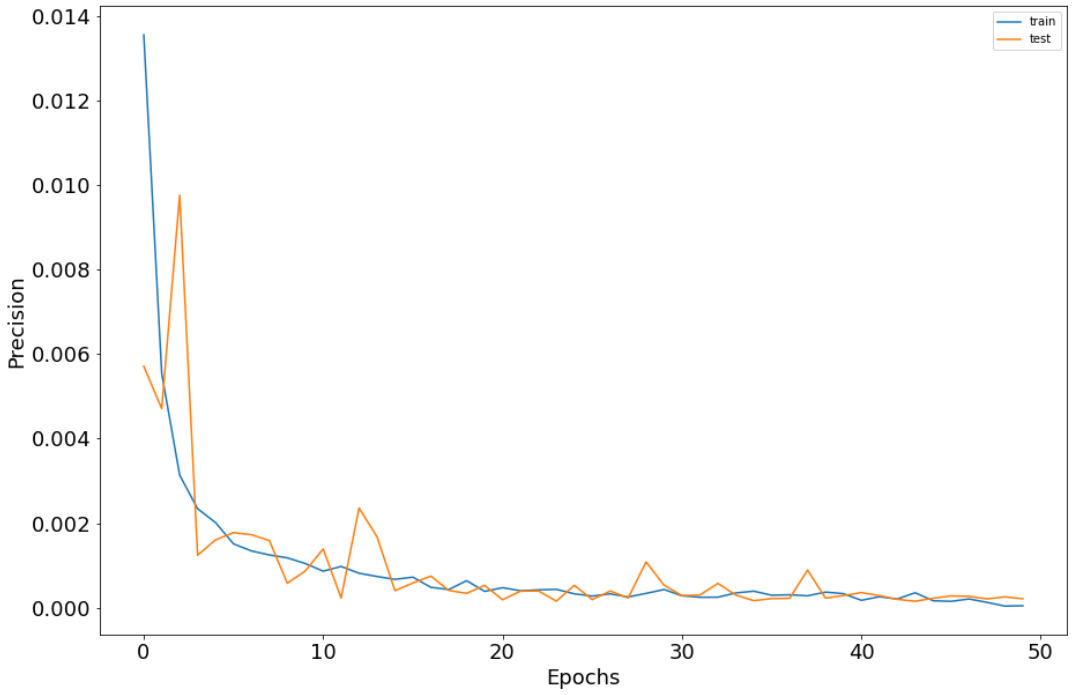


Figure 22. Precision values based on the two-week interval (CNN).

Recall score results look as following (Figure 23).

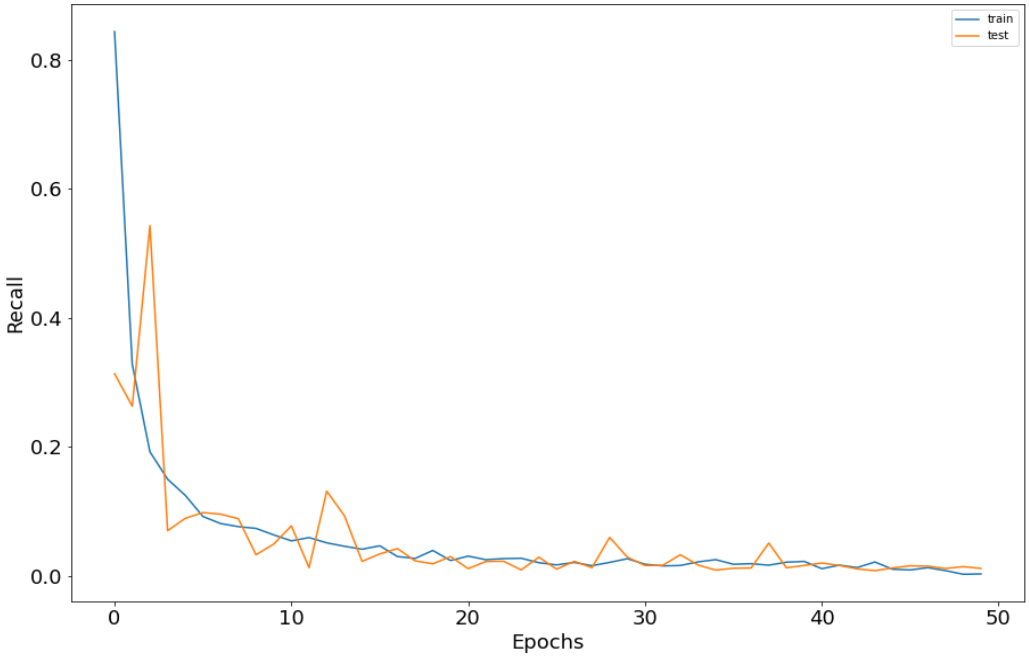


Figure 23. Recall values based on the two-week interval (CNN).

Results based on the narrower period are even less satisfactory, both recall and precision are equal to 0 while the model makes predictions.

LSTM-model was also validated with stratified 10-fold cross-validation and parameters were chosen based on the model performance on 10 folds. The result looks as following (Table 16).

Table 16. Cross-validation results based on the two-week interval (LSTM)

| **Mean accuracy** | **Mean precision** | **Mean recall** | **Mean F-score** |
|---|---|---|---|
| 0.99 | 0.97 | 0.7 | 0.82 |

As can be seen from the table, the recall score slightly dropped, moreover, the model has a different result of recall metric in folds, having a 0.8 recall score in some epochs, while also having 0.3 in others – therefore it can be concluded that the model prediction ability is not sufficient.

Then the model was used to make predictions based on the test data. The accuracy score results look as following (Figure 24).

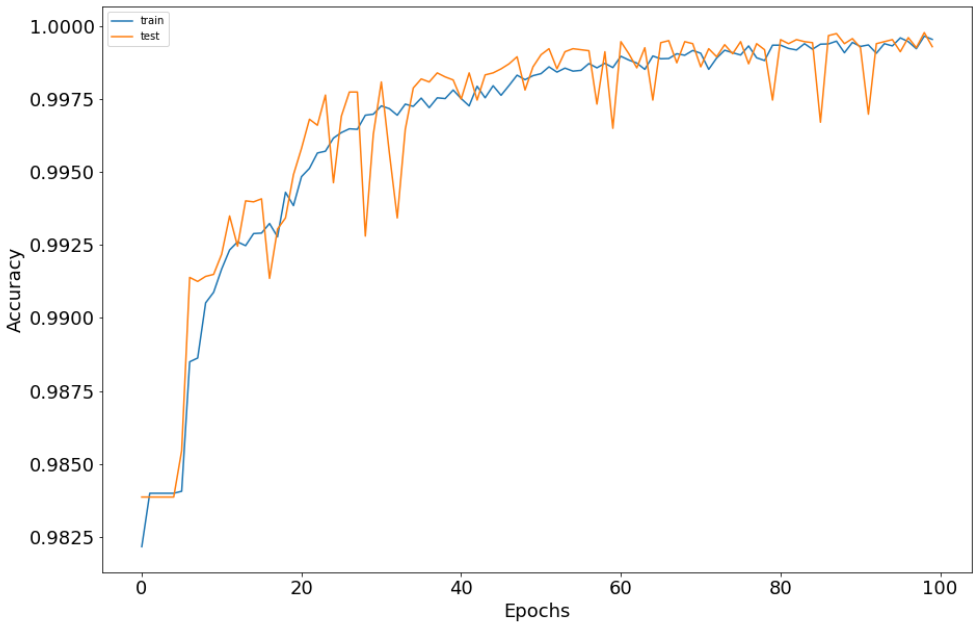


Figure 24. LSTM-model prediction accuracy based on a two-week interval.

On the test data, LSTM-model showed good results in terms of accuracy, it is seen that from the very beginning of the training accuracy is already high. Recall and precision scores were calculated for LSTM-model. The recall score results look as following (Figure 25).

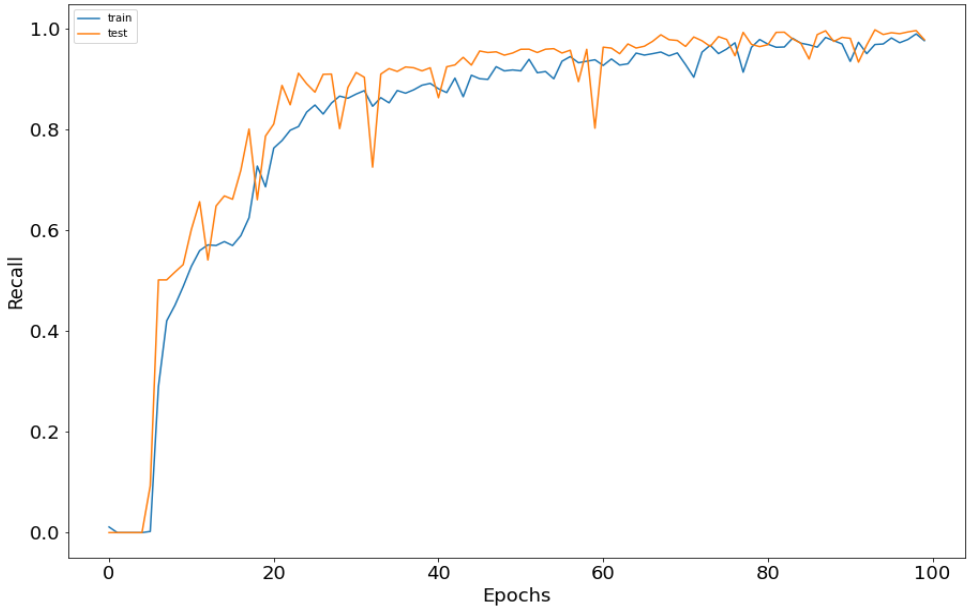


Figure 25. Recall values based on the two-week interval (LSTM).

The precision score results look as following (Figure 26).

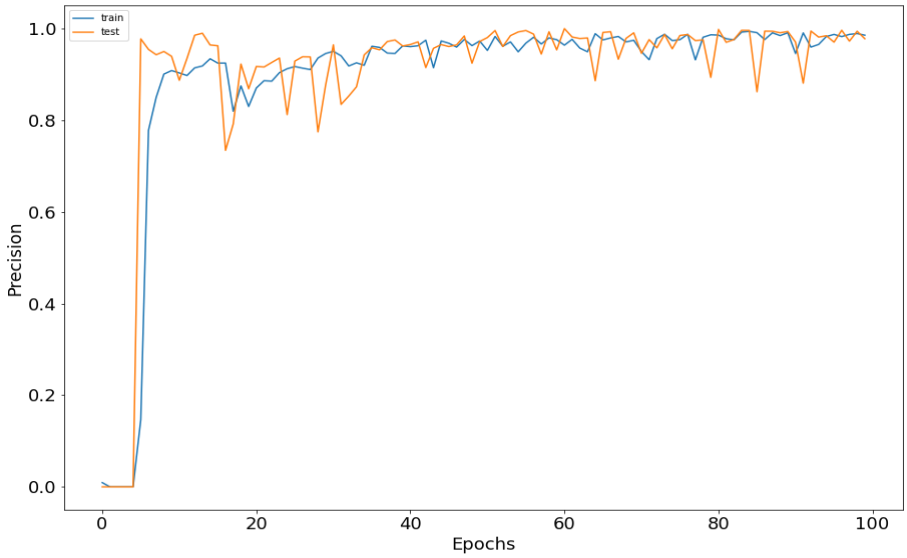


Figure 26. Precision values based on the two-week interval (LSTM).

In terms of precision and recall values, the Long-Short Term memory model shows good results even within the narrower period. Precision and recall values are equal to 1 after the 40th epoch approximately.

As can be seen from the models, the LSTM model is more accurate and precise than CNN even based on a narrower period.

## 6.5. Models comparison

6 machine learning models and 2 deep learning models were trained based on two periods considered as boundaries of the confidence interval of the decision-making process, models were validated using stratified 10-fold cross-validation. The model was tuned with selected hyperparameters to get better results within the making predictions stage, however, the change of hyperparameters did not significantly influence the results in those models that did not show before the ability to make predictions accurately.

Such machine learning models as Random Forest Classifier, Decision Tree Classifier, and KNN-classifier had four metrics (accuracy, precision, recall, f-score) equal to 1 on both validation and test data, meaning that they predict all samples correctly. Logistic Regression and SVM classifier presented high scores of accuracies and recall (almost equal to 1) on both validation and test data, though low results of precision, and, consequently, f-score, meaning that models are not able to predict true labels accurately. Naïve Bayes Classifier showed intermediate results, having recall and precision values equal to 0.77 and 0.82 respectively on the test data based on the four-week interval, and recall and precision scores equal to 0.26 and 0.97 on the test data based on the two-week interval, which means decreasing ability to identify correctly True Positives.

CNN-model did not present significant results based on both selected intervals, having high values of accuracy, its precision and recall metrics are next to 0. LSTM-model, on the contrary, presented good results in terms of four metrics on both validation and test data with a decrease in recall values within the cross-validation process.

Implementation of predictive maintenance should also consider the training time of the models; therefore, classifiers' training time was considered. The result looks as following (Table 17).

Table 17. The training time of the models

| **Model** | **Training time (min)** |
|---|---|
| Logistic regression | 0.8 |
| Support Vector Machine | 1 |
| Decision Trees | 1.1 |
| Random Forest | 1.2 |
| Naïve Bayes | 1 |
| KNN | 17 |
| CNN | 60 |
| LSTM | 70 |

As can be seen from the table, Logistic Regression took the smallest amount of time to train (48 seconds), CNN and LSTM models took the greatest amount of time to train (60 and 70 minutes respectively). KNN-classifier took moderate 17 minutes to train. However, the best results are presented by the Decision Tree Classifier and Random Forest Classifier with 1 minute approximately need to train, still being able to make predictions accurately.

Since the company is interested in the interpretable result, the Decision Tree Classifier and Random Forest classifier were chosen. Even though KNN-classifier and LSTM-model also showed promising results in terms of four selected metrics, they lack interpretability and extraction of decision rules which can be helpful for further improvement of the model or for implementation on other equipment in the company, moreover, training and testing KNN-classifier or LSTM-model took more time more than training and testing both selected models.

# 7. Managerial application of the research

Implementation and comparison of different models revealed differences between model prediction ability for imbalanced classification problems. While several algorithms can be used for solving such tasks as presented in the thesis, other models were deeply influenced by the presence of a big number of not-labelled instances. The research showed that the implementation of machine learning and deep learning algorithms can be used for predictive maintenance based on the data provided by the biotechnological company. Thus, for the industry, this means that additional research and initiatives should be conducted and involved.

Misbalanced data input influences many widely used models even while using cross-validation methods. Moreover, a comparison of models results within different time before maintenance need was conducted and revealed high accuracy of such models as Random Forest and Decision Tree.

The thesis can be used by the companies to estimate the remaining useful life of chromatographs` components and to implement predictive maintenance. Selected algorithms are highly influenced by the input data; therefore, these models can be useful for those situations when data consists of textual information and time-related features.

For many other problems, different specifications of models and formulations of problems are needed. For example, many devices also have data related to pressure, noise, vibration, etc. – data related to physical values. Therefore, further analysis can be done considering the addition of other physics-related data.

In terms of BIOCAD`s initiatives of implementation of predictive maintenance, several steps can be implemented. First, a function for data preprocessing in Python can be implemented. The function can take input data, follow the steps discussed in the thesis, transform it into the numerical form and apply machine learning or deep learning model. Moreover, an interactive dashboard can be invented for lamps and chromatographs operating time visualization. The dashboard example looks as following (Figure 27):

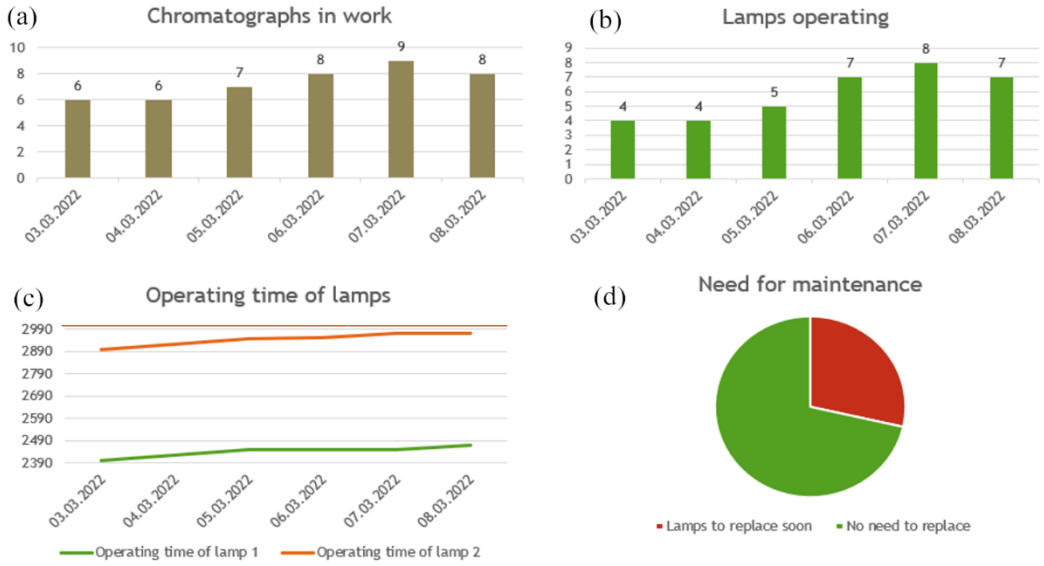


Figure 27. (a) Chromatographs in work chart; (b) Number of operating lamps chart; (c) Operating time of lamps chart; (d) Need for maintenance chart.

Recommendations for BIOCAD are the following:

1. To implement predictive maintenance algorithms in production. This will allow to reduce maintenance costs, avoid unexpected failures, and maximize equipment lifespan.
2. To choose either Decision Tree Classifier or Random Forest classifier as the main model for predictive maintenance. Both classifiers showed similar results both on validation and test data based on two intervals used for lamp replacement prediction. Accuracy, precision, recall, and F scores are equal to 1, which means that all cases were predicted accurately. Training time was the same for both models. However, the Random Forest classifier is a deeper and more difficult algorithm, though, it is less prone to overfitting than the Decision Tree Classifier.
3. Add information considering lamps` features which can enhance the input data, so that other models can be assessed

# Conclusion

The thesis presented an example of predictive maintenance implementation for the biotechnology company BIOCAD. It contributes to the shrinkage of the existing literature gap, particularly by describing the details of different models implementation for misbalanced and limited input data. Moreover, it implemented a real-life case study for an international biotechnology company, analysed the current company approach towards predictive maintenance implementation, and provided recommendations for further research and analysis.

Due to the increased interest of the global community in research in the field of medicine and biotechnology because of the pandemic, new processes, challenges, and opportunities will be faced by biotechnology and pharmaceutical companies. Their stakeholders expect the company to conduct research without any delays, loss of results, or any consecutive problems.

The industry is highly competitive and such methods as predictive maintenance algorithms provide a way to stand out. Nowadays the whole biotechnology industry is influenced by digitalization. Therefore, machine learning and deep learning models usage are becoming more and more important.

Despite the growing popularity of predictive maintenance algorithms, their usage is highly limited by the data quality and specificity of the situation. Usage of algorithms based on industrial data is discussed now to a limited degree.

From the scientific point of view the further research questions can be as follows:

1. How does the addition of input data influence the models' results?
2. Are these models applicable for predictive maintenance of other components or devices in the company?
3. How data preprocessing and calculations can be implemented into the daily operation process?

Models were divided into two parts: machine learning and deep learning algorithms. Each group was also divided into subgroups by the period of the prediction: within four-week or two-week intervals for assessing the need for lamp replacement. It provided the confidence interval that is

important for the company to be able to order and deliver lamps without any delays and requirements of extra costs and storage.

To achieve the goals presented in the introduction, in this research classification algorithms that allowed to make predictions and compare their results in terms of four metrics: accuracy, precision, recall, and f-score on the given imbalanced data were introduced.

The managerial conclusions were mostly dedicated to the importance of predictive maintenance implementation to the industry. Poor predictive maintenance algorithms can highly influence the company's operating process and damage the company`s brand.

The thesis results answer the stated research questions:

1. Existing machine learning and deep learning models can be used for the analysis of sequential data. Decision Tree Classifier, Random Forest Classifier, and KNN-classifier showed the best results in terms of precision and recall values among ML models, LSTM-model outperformed CNN model. Decision Tree Classifier and Random Forest Classifier are the most interpretable models, and their training does not take a great amount of time which is crucial for industry implementation.
2. Different models provide unequal results. They are highly influenced by the data quality and general structure of the data. Therefore, a comprehensive analysis of each situation must be performed.
3. Since the data has outliers, several models showed unstable results within a narrower period of prediction. Several models such as the Decision Tree Classifier, Random Forest Classifier, and KNN-classifier showed the same level of accuracy, precision, and recall even within a narrower period. The results can be improved by using a larger dataset for model training.

In the end, despite the thesis being one of the few dedicated to the complex topic of predictive maintenance in the biotechnological industry, the stated research goals were achieved. The research included an in-depth analysis and comparison of models that can be applied to the data provided by BIOCAD. Decision rules used by Decision Tree Classifier were discussed and interpreted, they revealed the uncertainty created by the presence of the outliers in the data. Moreover, high-performance measures can be influenced by the increase in the replacement rate. Therefore, future investigations of outliers' effect on the data and analysis of the number of target values are necessary to validate the conclusions that can be drawn from this study. The identified

challenges are the lack of interpretability of such models as the KNN-classifier and LSTM-model. Though they showed promising results for predictive maintenance implementation, the lack of explanations of their predictions based on the industrial data keeps a branch of opportunities for future research in the topic.

# References

Alloghani M., Dhiya A-J, Mustafina J., Hussain A., Ahmed J. A. (2020) A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science. *Springer International Publishing* doi:10.1007/978-3-030-22475-2_1.

Altman N. (1992) An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician.* 46. 175–185. doi:10.1080/00031305.1992.10475879. hdl:1813/31637.

Ayvaz S., Koray A. (2021) Predictive maintenance system for production lines in manufacturing: A machine learning approach using IoT data in real-time. *Expert systems with applications.* doi: 10.1016/j.eswa.2021.114598.

Baru A. (2018) Three Ways to Estimate Remaining Useful Life for Predictive Maintenance - MATLAB & Simulink. [online] Available at: https://se.mathworks.com/company/newsletters/articles/three-ways-to-estimate-remaining-useful-life-for-predictive-maintenance.html (Accessed: 15 March 2022)

Bescond P-L. (2020) Cyclical features encoding, it's about time! [online] Available at: https://towardsdatascience.com/cyclical-features-encoding-its-about-time-ce23581845ca (Accessed: 19 February 2022)

Beucher A., Rasmussen C., Moeslund T., Greve M. (2022) Interpretation of Convolutional Neural Networks for Acid Sulfate Soil Classification Frontiers in Environmental Science. [online] Available at: https://www.frontiersin.org/article/10.3389/fenvs.2021.809995 (Accessed: 7 February 2022)

BIOCAD. (2019/2020) Report on sustainable development.

BIOCAD. [online] Available at: https://biocad.ru/science (Accessed: 2 February 2022)

Black Paul E. (2019) Manhattan distance. Dictionary of Algorithms and Data Structures. [online] Available at: https://xlinux.nist.gov/dads/HTML/manhattanDistance.html (Accessed: 27 February 2022)

Borst N. (2020) Adaptations for CNN-LSTM Network for Remaining Useful Life Prediction: Adaptable Time Window and Sub-Network Training. *Master's Thesis, Delft University of Technology.*

Bousdekis A., Lepenioti K., Apostolou D., Mentzas G. (2019) Decision Making in Predictive Maintenance: Literature Review and Research Agenda for Industry 4.0. *IFAC-PapersOnLine.* 52. 13. (2019). 607-612. ISSN 2405-8963. doi:10.1016/j.ifacol.2019.11.226.

Capozzoli A., Cerquitelli T., Piscitelli M. (2016) Chapter 11 - Enhancing energy efficiency in buildings through innovative data analytics technologies. *Academic Press.* 353-389. doi:10.1016/B978-0-12-803663-1.00011-5.

Carvalho T., Soares F., Vita R., Francisco R., Basto J., Alcalá S. (2019) A systematic literature review of machine learning methods applied to predictive maintenance. *Computers & Industrial Engineering.*137. doi: 10.1016/j.cie.2019.106024.

Chekkala V. L. (2020) Predictive Maintenance for Fault Diagnosis and Failure Prognosis in Hydraulic System. *Master`s Thesis. National College of Ireland.*

Ciresan D. C., Ueli M., Jonathan M., Gambardella L., Schmidhuber J. (2011) Flexible, High Performance Convolutional Neural Networks for Image Classification. *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence.* 2. 1237–1242.

Coelho D., Costa D., Rocha E., Almeida D, Santos J. (2022) Predictive maintenance on sensorized stamping presses by time series segmentation, anomaly detection, and classification algorithms. *Procedia Computer Science.* 200. doi:10.1016/j.procs.2022.01.318.

Cohen D. (2004) Precalculus: A Problems-Oriented Approach. *Cengage Learning.* 698.

Dangeti P. (2017) Statistics for Machine Learning: Techniques for exploring supervised, unsupervised, and reinforcement learning models with Python and R. *Packt Publishing.*

Efthymiou K., Papakostas N., Mourtzis D., Chryssolouris G. (2012) On a predictive maintenance platform for production systems. ProcediaCIRP. 3. 221–226. doi:10.1016/j.procir.2012.07.039.

Goodfellow I., Bengio Y., Courville A. (2016) Deep learning. *MIT press*

Goodfellow I., Bengio Y., Courville A. (2016) Softmax Units for Multinoulli Output Distributions. Deep Learning. *MIT Press*. 180–184.

Hassan I. F., Forestier G., Weber J., Idoumghar L., Muller P-A. (2019) Deep learning for time series classification: a review. *Springer International Publishing.*

He H., Ma Y. Imbalanced learning: foundations, algorithms, and applications. (2013) *Wiley.*

Herrera F., Charte F., Antonio. J R., Jesus M. (2016) Multilabel Classification Problem Analysis, Metrics and Techniques. *Springer International Publishing*. doi:10.1007/978-3-319-41111-8

Hochreiter S., Schmidhuber J. (1997) Long short-term memory. *Neural computation.* 9. 1735-1780 doi: 10.1162/neco.1997.9.8.1735.

Hossein R. M., Khushefati W. H., Demirboga R., Adeli Hojjat. (2017) Supervised deep restricted Boltzmann machine for estimation of concrete. *ACI Materials Journal*. 114(2). 237–244.

Hu T., Guo Y., Gu L., Zhou Y., Zhang Z., Zhou Z. (2022) Remaining useful life prediction of bearings under different working conditions using a deep feature disentanglement-based transfer learning method. *Reliability Engineering & System Safety*. doi:10.1016/j.ress.2021.108265.

Jardine A. K. S, Daming L., Dragan B. (2006) A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mech. Syst. Sig. Process.* 20. 1483–1510.

Kang Z., Catal C., Tekinerdogan B. (2021) Remaining Useful Life (RUL) Prediction of Equipment in Production Lines Using Artificial Neural Networks. *Sensors.* 21. doi: 10.3390/ s21030932

Keyur V. (2020) Why a run-to-fail strategy is not good practice for SIS? IOP *Conference Series: Materials Science and Engineering.*

Konda S., Kavitha R., Varaprasad R. M., Madhukar G., Venkata R. (2019) Convolution Neural Networks for Binary Classification. *Journal of Computational and Theoretical Nanoscience*. 16. 4877-4882. doi:10.1166/jctn.2019.8399.

Leohold S., Engbers H., Freitag M. (2021) Prognostic Methods for Predictive Maintenance: A generalized Topology. *IFAC-PapersOnLine*. 54. Issue 1. 629-634.

Liang X., Jun X. (2021) Biased ReLU neural networks. *Neurocomputing.* Volume 423. 71-79. ISSN 0925-2312. doi:10.1016/j.neucom.2020.09.050.

MathWorks. (2016) Applying supervised learning. Machine Learning with MATLAB.

Mikulic M. (2021) Pharmaceutical market: worldwide revenue 2001-2020. [online] Available at: https://www.statista.com/statistics/263102/pharmaceutical-market-worldwide-revenue-since-2001/ (Accessed: 2 March 2022)

Mishra M. (2018) Prognostics and Health Management of Engineering Systems for Operation and Maintenance Optimisation. *Ph.D. Thesis. Lulea University of Technology.*

Montero J. J., Schwartz S., Vingerhoeds R., Grabot B., Salaün M. (2020) Towards multi-model approaches to predictive maintenance: A systematic literature survey on diagnostics and prognostics. *Journal of Manufacturing Systems.* 56. 539–557. doi: 10.1016/j.jmsy.2020.07.008

Mossotto E., Ashton J. J., Coelho T., Beatti M., MacArthur B., Ennis S. (2017) Classification of paediatric inflammatory bowel disease using machine learning. *Scientific Reports.* 7. 1–10.

Naik N., Purohit S. (2017) Comparative Study of Binary Classification Methods to Analyze a Massive Dataset on Virtual Machine. *Procedia Computer Science.* Volume 112. 1863-1870. ISSN 1877-0509

Ng S., Xing Y., Kwok L.T. (2014) A naive Bayes model for robust remaining useful life prediction of lithium-ion battery. *Applied Energy*. 118. 114-123. doi:10.1016/j.apenergy.2013.12.020

Nikfar M., Bitencourt J., Mykoniatis K. (2022) A Two-Phase Machine Learning Approach for Predictive Maintenance of Low Voltage Industrial Motors. *Procedia Computer Science*. 200. 111-120. doi:10.1016/j.procs.2022.01.210.

Ogunmolu O., Gu X., Jiang S., Gans N. (2016) Nonlinear Systems Identification Using Deep Dynamic Neural Networks. arXiv:1610.01439

Patel P. (2018) Why Python is the most popular language used for Machine Learning. [online] Available at: https://medium.com/@UdacityINDIA/why-use-python-for-machine-learning-e4b0b4457a77 (Accessed: 27 February 2022)

Project Services Co. – Qatar. (2017). [online] Available at: https://www.pscqatar.com/index.php?page=display_part&pid=138 (Accessed: 23 March 2022)

PwC. (2016) Industry 4.0: building the digital enterprise. Technical report. [online] Available at: www.pwc.com/industry40. (Accessed: 10 March 2022)

PwC. (2018) Predictive Maintenance 4.0. Beyond the Hype: PdM 4.0 Delivers Results

Qiu J., Wu Q., Ding G., Xu Y., Shuo F. (2016) A survey of machine learning for big data processing. *EURASIP Journal on Advances in Signal Processing*. 1–16.

RBK Group. Govoryat, chto biotechnologii – novoye programmirovaniye. [online] Available at: https://biocad-it.rbc.ru/?fbclid=IwAR231fZQfOZ51RYwosZ31DlR6Av97bnr0ROWpmtLNGyeq56Jf-Hm0seCk-c (Accessed: 18 March 2022)

Samsudin R., Ani S., Saad P. (2010) A Comparison of Time Series Forecasting using Support Vector Machine and Artificial Neural Network Model. *Journal of Applied Sciences*. 10: 950-958.

Sean O. (2019) The Case for Predictive Maintenance. [online] Available at: https://www.pharmamanufacturing.com/articles/2019/the-case-for-predictive-maintenance/ (Accessed: 12 March 2022)

Sharma N. (2019) Importance of Distance Metrics in Machine Learning Modelling. [online] Available at: https://towardsdatascience.com/importance-of-distance-metrics-in-machine-learning-modelling-e51395ffe60d (Accessed: 10 March 2022)

Sherstinsky A. (2020) Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network. Elsevier.

Sniderman B., Mahto M., Cotteleer M. (2016) Industry 4.0 and manufacturing ecosystems: Exploring the world of connected enterprises. *Deloitte 620 Consulting.*

Sotiris K. (2007) Supervised machine learning: A review of classification techniques. *Informatica.* 31. 249–268.

Staples L. (2018) A Comparison of the Predictive Ability of Logistic Regression and Time Series Analysis on Business Credit Data. Grey Literature from Ph.D. Candidates. 9. [online] Available at: https://digitalcommons.kennesaw.edu/dataphdgreylit/9 (Accessed: 25 February 2022)

Szandała T. (2018) Review and Comparison of Commonly Used Activation Functions for Deep Neural Networks. arXiv. doi: 10.48550/arXiv.2010.09458

Tao F., Qi Q., Ang L., Kusiak A. (2018) Data-driven smart manufacturing. 48. *Elsevier*. 157–69. doi:10.1016/j.jmsy.2018.01.006.

Thanaki J. (2017) Python Natural Language Processing. *Birmingham*. 120-140.

Tiago Z., Cristiano A., Felipe A., Ramos G., Kunst R., Righi R. (2022) A predictive maintenance model for optimizing production schedule using deep neural networks. *Journal of Manufacturing Systems.* 62450-462 doi:10.1016/j.jmsy.2021.12.013.

Tiddens W. (2018) Setting sail towards predictive maintenance developing tools to conquer difficulties in the implementation of maintenance analytics. *Ph.D. thesis. The University of Twente.*

Tolles J., Meurer W. J. (2016) Logistic Regression Relating Patient Characteristics to Outcomes. *JAMA.* 316 (5). 533–4. doi:10.1001/jama.2016.7653.

Wang B., Lei Y., Tao Y., Li N., Liang G. (2020) Recurrent convolutional neural network: A new framework for remaining useful life prediction of machinery. *Neurocomputing.* 117–129.

Wehle H. (2017) Machine Learning, Deep Learning, and AI: What's the Difference? [online] Available at: https://www.freecodecamp.org/news/ai-vs-ml-whats-the-difference/#:~:text=The%20Difference%20Between%20AI%20and,all%20AI%20is%20machine%20learning. (Accessed: 10 February 2022)

Wu D., Connor J., Terpenny J., Gao R., Kumara S. (2017) A Comparative Study on Machine Learning Algorithms for Smart Manufacturing: Tool Wear Prediction Using Random Forests. *Journal of Manufacturing Science and Engineering*. 139 (7). 071018. doi:10.1115/1.4036350

Wu J., Kui H., Cheng Y., Zhu H., Shao X., Yuan H. (2020) Data-driven remaining useful life prediction via multiple sensor signals and deep long short-term memory neural network. *ISA Trans*. 241–250

Yurek O., Birant D. (2019) Remaining Useful Life Estimation for Predictive Maintenance Using Feature Engineering. *2019 Innovations in Intelligent Systems and Applications Conference*. doi: 10.1109/ASYU48272. 8946397.

Zhai S., Gehring B., Reinhart G. (2021) Enabling predictive maintenance integrated production scheduling by operation-specific health prognostics with generative deep learning. *Journal of Manufacturing Systems.* 61. 830-855. doi: 10.1016/j.jmsy.2021.02.006.

Zhang W., Dong Y., Wang H. (2019) Data-driven methods for predictive maintenance of industrial equipment: A survey. *IEEE Systems Journal*. 13(3). 2213–2227. doi:10.1109/JSYST.2019.2905565.

Zhang X., Wang D. (2016) A deep ensemble learning method for monaural speech separation. *IEEE/ACM. Transactions on Audio, Speech, and Language Processing*. 24 (5). 967–977.

Zhao R., Wang D., Ruqiang Y., Kezhi M., Fei S., Wang J. (2017) Machine health monitoring using local feature-based gated recurrent unit networks. *IEEE Trans Ind Electron* 65:1539–48. doi:10.1109/TIE.2017.2733438