Saint Petersburg State University Graduate School of Management

Master of Business Analytics and Big Data

DATA-DRIVEN APPROACH FOR THE PREVENTIVE MAINTENANCE OF CHROMATOGRAPHS

Consulting project for BIOCAD

Master's Thesis by the 2nd year students Concentration — BM.5783.2019

Master in Business Analytics and Big Data

Elizareva Natalia

Titova Diana

Research Advisor: Elvira V. Strakhovich

Ass. Professor, Information Technologies in Management Department

Saint Petersburg 2022

ЗАЯВЛЕНИЕ О САМОСТОЯТЕЛЬНОМ ХАРАКТЕРЕ ВЫПОЛНЕНИЯ

ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ

Мы, Титова Диана и Елизарьева Наталья, студентки второго курса магистратуры направления «Бизнес-аналитика и большие данные», заявляем, что в нашей магистерской диссертации на тему «Превентивное обслуживание хроматографов на основе данных», представленной в службу обеспечения программ магистратуры для последующей передачи в государственную аттестационную комиссию для публичной защиты, не содержится элементов плагиата.

Все прямые заимствования из печатных и электронных источников, а также из защищенных ранее выпускных квалификационных работ, кандидатских и докторских диссертаций имеют соответствующие ссылки.

Нам известно содержание п. 9.7.1 Правил обучения по основным образовательным программам высшего и среднего профессионального образования в СПбГУ о том, что «ВКР выполняется индивидуально каждым студентом под руководством назначенного ему научного руководителя», и п. 51 Устава федерального государственного бюджетного образовательного учреждения высшего образования «Санкт-Петербургский государственный университет» о том, что «студент подлежит отчислению из Санкт-Петербургского университета за представление курсовой или выпускной квалификационной работы, выполненной другим лицом (лицами)».

30.05.2022  Титова Диана

30.05.2022  Елизарьева Наталья

STATEMENT ABOUT THE INDEPENDENT CHARACTER OF

THE MASTER THESIS

We, Titova Diana and Elizareva Natalia, second year master students of the program «Business analytics and the Big Data», state that my master thesis on the topic «Data-driven approach for the preventive maintenance of chromatographs», which is presented to the Master Office to be submitted to the Official Defense Committee for the public defense, does not contain any elements of plagiarism.

All direct borrowings from printed and electronic sources, as well as from master theses, PhD and doctorate theses which were defended earlier, have appropriate references.

We are aware that according to paragraph 9.7.1. of Guidelines for instruction in major curriculum programs of higher and secondary professional education at St.Petersburg University «A master thesis must be completed by each of the degree candidates individually under the supervision of his or her advisor», and according to paragraph 51 of Charter of the Federal State Institution of Higher Education Saint-Petersburg State University «a student can be expelled from St.Petersburg University for submitting of the course or graduation qualification work developed by other person (persons)».

30.05.2022  Титова Диана

30.05.2022  Елизарьева Наталья

Аннотация

Авторы	Елизарьева Наталья Титова Диана
Название магистерской диссертации	Превентивное обслуживание хроматографов на основе данных
Факультет	Высшая школа менеджмента
Направление подготовки	Бизнес-аналитика и большие данные
Год	2022
Научный руководитель	Страхович Эльвира
Описание цели, задач и основных результатов	Существует много работ, посвященных внедрению прогностического технического обслуживания. Тем не менее, качество данных сильно влияет на процесс реализации модели, а бизнес-кейсы, посвященные этому явлению, ограничены. В этой магистерской диссертации представлен анализ внедрения моделей профилактического обслуживания в биотехнологическую промышленность. В диссертации рассматриваются два подхода к расчету остаточного срока использования. Результаты были использованы для построения прогностической модели и использованием пакета Prophet. Описываемые методы расчета рабочего времени обеспечивают компании модели, которые можно использовать в работе. Кроме того, были представлены бизнес-рекомендации. На основе выводов даны управленческие рекомендации.
Ключевые слова	превентивное обслуживание, биотехнологии, замена, модель временного ряда

Annotation

Master Student Names	Elizareva Natalia Titova Diana
Master Thesis Title	Data-driven approach for the preventive maintenance of chromatographs
Title Faculty	Graduate school of Management
Main field of study	Business Analytics and Big Data
Year	2022
Academic Advisor`s Name	Strakhovich Elvira
Description of the goals, tasks and main results	There are many works considering prognostic maintenance implementation. Nevertheless, input data highly influences the process of model implementation and business cases devoted to this phenomenon are limited. This Master's Thesis provides an analysis of preventive maintenance models implementation to the biotechnological industry. In the literature review main directions of preventive maintenance research and approaches to remaining useful life estimation were discussed. In the empirical part of the thesis two approaches for the remaining useful life calculations are observed. Results are used to build a forecast model using the Prophet package. Defined methods of working hours calculation provide the company with the models that can be used in the industry. Moreover, business recommendations are provided. Based on the findings, managerial implications are given.
Keywords	preventive maintenance, biotechnology, replacement, time-series model

List of abbreviations

PdM Predictive Maintenance

PM Preventive Maintenance

IoT Internet of things

ML Machine Learning

RUL Remaining Useful Life

CMMS Computerized Maintenance Management Systems

CAD Computer Aided Design

PLM Product Lifecycle Management

MES Manufacturing Execution Systems

EAM Enterprise Asset Management

FSM Field Service Management

ERP Enterprise resource planning

HR Human Resource

AI Artificial Intelligence

SaaS Software as a Service

OECD Organisation for Economic Co-operation and Development

IP Intellectual Property

MAE Mean Absolute Error

MAPE Mean Absolute Percentage Error

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. Introduction

1.1. Background

With the rapid development of modern technology, people invent new ways to improve industrial processes. The term Industry 4.0, sometimes referred to as smart industry, has become widespread. It means the application of automated and modern data analysis models to the decision-making process. The concept can be divided into the following segments: Smart Manufacturing, Internet of Things (IoT), Smart Factory, and Unattended Manufacturing. Preventive maintenance is an example of Industry 4.0 implementation.

The boost of the preventive maintenance techniques can be explained by the pressure that process manufacturers have been experiencing in recent years because resources are becoming more expensive, and the growth of business have slowed to a crawl. Productivity growth for industrial companies in the European Union fell from an average of 2.9 percent over the 1996–2005 period to just 1.6 percent from 2006–2015, according to the OECD (McKinsey&Co, 2017). Most businesses have already made many changes to their business models to optimize all the operations including supply chain management. So now companies face the urgent necessity to find new ways to increase their productivity and profitability of their operations. One of the significant assets that have not been yet optimized is the company's data.

Despite the world-wide implementation of the preventive maintenance techniques, it still represents a great challenge for companies. It's not enough to just apply the technology to get good results. The industry 4.0 transformation requires to carefully blend traditional approaches and new tools, to combine business and process activities and to tailor them to specific features of the business (McKinsey&Co, 2017).

Current predictive maintenance (PdM) solutions can be divided into three broad categories (Zhang, et al., 2019):

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

Data-driven models can be classified in sensor-based, logbased, and hybrid approaches. Sensor-based approaches make use of time-series signals of single or multiple sensors without physical models to assess the remaining useful life (RUL). Sensor-based approaches are often applied in mechanical engineering, mainly on rotatory machines or components such as bearings or gear-boxes. Log-based approaches use historical event-log data (instead of sensor data) to train machine learning algorithms. RUL estimation is implemented by a predetermined level of failure probability and applying the model on real-time event-log data. Event-log data can either be aggregated data from sensor streams or extracted data from log messages like system messages, alarm codes, numerical values, or keywords. (Clemens Gutschi et al.)

Preventive maintenance is aimed to identify possible malfunctions ahead of time, which allows processes to not meet any suspensions and pauses in production. Preventive maintenance algorithms monitor and collect data, compare it with conditional baselines and detect any abnormalities or specific parameters. If there is a trigger, PM 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. For these reasons, preventive maintenance is now widely used in many spheres, and companies are actively investing in development of these prognostic models.

1.2. Research questions

The purpose of this paper is to provide an in-depth study of the application of preventive maintenance based on data from BIOCAD.

The work focuses on the following research questions:

1. What are the applicable models or approaches for preventive maintenance?
2. Is there an existing approach that can be used for the data provided by BIOCAD?
3. How accurately could the working hours and replacement time be calculated using existing or new approaches in terms of percentage error?

1.3. Model implementation

Usage of programming language will allow us to find answers to the research questions posed. The first task is to pre-process data and accurately calculate the working hours of the chromatographs` data provided by BIOCAD. Automated functions can be used for that purpose. The second task is to predict the operating time, which can be used with the help of machine learning (ML) algorithms. Supervised ML models use labelled data to find patterns which can be helpful for prognostic goals. These models can be represented by, for example, classification or regression tasks. It is going to be a regression task: the task is to create a prognostic model to calculate the operating time of the lamps. Therefore, a regression model will be used.

1.4. Structure of the thesis

As part of the graduation thesis, the mechanics of preventive maintenance implementation are first explained. The second part of the paper is a literature review of important articles on preventive maintenance and different methods of estimating remaining useful life. Then, we will focus on the preventive maintenance mechanisms and the specifics of preventive maintenance in biotech companies. Chapter 5 explains the usage of the methodology used in the paper, advantages of ML model`s usage. Then two approaches created for operating hours ‘calculation are discussed and compared, forecast results are observed. Functions and the models are created and executed using the Python programming language.

1.5 Project plan

1.5.1 Project objectives

We have identified the following objectives for completing the research (Table 1).

Table 1. Overview of the projects` goals.

Objective №	Objective description	Success criteria
1	Formulate specificity of the biotechnological industry situation and preventive maintenance implementation aspects.	The topic is discussed in the thesis` chapter.
2	Conduct exploratory data analysis for the data provided by BIOCAD. Check missing data.	The results are discussed, visualised and presented in the thesis` chapter.
3	Create and discuss approaches that are suitable for working hours calculations.	Created approaches are approved by BIOCAD`s representatives. General logic is presented in the thesis ‘chapter.
4	Calculate working hours based on the first approach. Choose a suitable metric for error rate analysis.	Working hours calculations for all lamps are presented in the code sent to the company, results are visualised, suitable metric is chosen and implemented to calculate error rate. Applicability of the method is discussed in the thesis ‘chapter.
5	Calculate working hours based on the second approach. Choose a suitable metric for error rate analysis.	Working hours calculations for all lamps are presented in the code sent to the company, results are visualised, suitable metric is chosen and implemented to calculate error rate. Applicability of

		the method and comparison with the first approach are discussed in the thesis ‘chapter.
6	Build the regression model, evaluate the model`s accuracy.	A forecast of lamps` working hours is created. The model is evaluated with the suitable metrics.
7	Choose appropriate visualisations to represent information to the BIOCAD`S specialists. Draw managerial recommendations from the conducted analysis.	Presented results are approved by BIOCAD`s representatives.

Source: [authors research]

1.5.2 Project team

The following table represents the main participants in the thesis, information about their roles and contact details (Table 2).

Table 2. Project Team

Name	Role	Email
Natalia Elizareva	Researcher	st085838@gsom.spbu.ru
Diana Titova	Researcher	st057275@gsom.spbu.ru
Elvira Strakhovich	Academic Supervisor	strakhovich@gsom.spbu.ru
Vologdin Vasilii	Biocad representative	vologdin@biocad.ru
Nina Khabarova	Biocad representative	khabarovana@biocad.ru

Source: [authors research]

1.5.3 Research framework

We have used CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology to construct the research framework (Table 3).

Table 3. Project goals within CRISP-DM methodology.

Objective(s) №	CRISP-DM phase	Chapter
1	Business understanding	4.1. Pharmaceutical companies` specifics of preventive maintenance implementation 4.2. Company overview 4.3 Company's data pipeline overview
2	Data understanding Data preparation	6.1. Input data
3	Modelling Evaluation	6.2. Approaches to RUL estimation based on the BIOCAD`s data
4		6.3. Calculation of the working hours based on the first approach.
5		6.4. Calculation of the working hours based on the first approach.
6	Regression model building Evaluation	6.3 Calculation of the working hours based on the first approach.

		6.4. Calculation of the working hours based on the first approach.
7	Preparation of the final report	6.6. Dashboard preparation

Source: [authors research]

2. Literature review

Basically, the literature used in the work can be divided into two main categories: the main directions of preventive maintenance research and the methods of remaining useful life estimation. Smart industrial information and theoretical materials considering model application and data pre-processing methods are also used.

There are many works dealing with the impact of Industry 4.0 or Smart Industry on different branches. For example, in a work titled "Implementation of Industry 4.0: Sustainability Relevance and Potential Social Impacts in Developing Countries", Walter Cardoso Satyro considered the implications of this process, identifying the relative strengths and weaknesses of the implementation process, and evaluates the relevance of sustainable development in the smart industry and its social impact analysis.

2.1 Main directions of preventive maintenance research.

Machinery maintenance service can be subdivided into 2 groups: unplanned and planned. The first one (unplanned or run-to failure maintenance) is a naïve and intuitive approach. The logic of this method is simple and straightforward: it is necessary to repair the machine when it breaks down. This methodology has been used as the base of the maintenance management system for many years since the first manufacturing plants were built. (Mobley, 2002)

However with time it became clear that the unplanned maintenance led to unforeseen shutdowns, decreased the productivity, efficiency, the quality of both products and processes, and the safety of industrial environments while increasing the costs related to maintenance interventions and spare parts supply and management. Also it became clear that the use of this type of maintenance

makes it impossible to develop long-term manufacturing strategies and establish long-term partnerships with customers. These consequences resulted in the necessity to transfer to the planned maintenance.

The term “planned maintenance” includes a broad set of approaches that are aimed to predict the machine’s necessity for maintenance and perform it right in time or before the machine breaks down.

Different sources name different types of planned maintenance which will be discussed later in this chapter. According to the ISO standard there are 3 types of planned maintenance:

1) condition-based maintenance (which was called the “predictive maintenance” before) is the maintenance performed as governed by condition monitoring programmes.

2) preventive maintenance - maintenance performed according to a fixed schedule, or according to a prescribed criterion, that detects or prevents degradation of a functional structure, system or component, in order to sustain or extend its useful life;

3) reliability centred maintenance(RCM) - disciplined logic used to identify those cost effective and technologically feasible maintenance tasks that realise the inherent reliability of equipment at a minimum expenditure of resources over the life of the equipment. (ISO)

The first step towards the planned maintenance was the task to find the optimal time interval between two consecutive services that would prevent the machine from breaking down within this period. This approach is called preventive maintenance. Preventive management program assumes that machines have the pre-set time period for degradation which is typical for their type. This time period creates the base for time interval calculation between two consecutive services. The preventive maintenance has great advantage over the unplanned management as it reduces unforeseen shutdowns and keep at the certain level the quality of products and processes. However such planned maintenance may not allow the exploitation of the whole life of components, which are replaced according to the schedule regardless of their actual health condition (Francesca Calabrese, 2021). It means that in the paradigm of the preventive maintenance the service action will be taken earlier and more often than it’s necessary, which in turn leads to growing expenditures on the maintenance management program.

Another problem is the possibility of preliminary equipment failure. As the preventive maintenance considers that the necessary time interval between 2 services is typical for all the

machines of the similar category, it ignores a lot of important parameters such as the quality of the equipment, specific working conditions, quality of the maintenance procedures previously realised etc. All these features may lead to the preliminary failure of the equipment. In this case, for example, if the pump fails before the end of the warranty period it must be repaired using run-to-failure techniques. Analysis of maintenance costs has shown that repairs made in the unplanned mode are normally three times greater than the same repairs made on a scheduled basis. (Mobley, 2002)

The most advanced technique is the predictive maintenance. Predictive maintenance is a philosophy or attitude that uses the actual operating condition of plant equipment and systems to optimise total plant operation. The base of this approach is the regular monitoring of actual operating conditions, operating efficiency, various machinery parameters such as vibration level, temperature. This monitoring provides the data required to analyse the actual state of the machine to ensure the maximum possible interval between repairs and to minimise the number and cost of unscheduled outages created by machine failures. (Mobley, 2002) The possibility of predictive maintenance existence and development is closely related to the accumulation of big data of equipment, historic activity in different industries, development of big data analysis methodologies and development of data storage infrastructure. The purpose of predictive maintenance is to minimize unscheduled equipment failures, maintenance costs, and lost production. It is also intended to improve the production efficiency and product quality in the plant.

Because of the great practical value, the topic of the predictive maintenance is of great interest not only for academic researchers and manufacturing companies but also for IT giants whose specialists also investigate planned maintenance and make their contribution to the development of maintenance management practices. For example, researchers from IBM identify 4 major types of preventive maintenance. Each is built around the concept of planned maintenance, although they are all organised and scheduled differently, to suit different business operation purposes:

1. Usage-based preventive maintenance is triggered by the actual utilisation of an asset. This type of maintenance takes into account the average daily usage or exposure to environmental conditions of an asset and uses it to forecast a due date for a future inspection or maintenance task.

2. Calendar/time-based preventive maintenance occurs at a scheduled time, based on a calendar interval (weekly, monthly etc).

3. Predictive maintenance is a more advanced planned maintenance technique based on the equipment condition in order to estimate when maintenance should be performed.

4. Prescriptive maintenance helps analyse and determine different options and potential outcomes, in order to mitigate any risk to the operation. (IBM)

Specialists from the Oracle company have also created the book describing different approaches to the planned maintenance management and their results. According to their studies, the predictive maintenance program can reduce equipment breakdowns by 70%-75%, reduce maintenance costs by 25%-30%, reduce downtime by 35%-45%, and increase production by 20%-25%. (Oracle, 2020)

Researchers from the Oracle company also cites the company's maintenance maturity model (Figure 1):

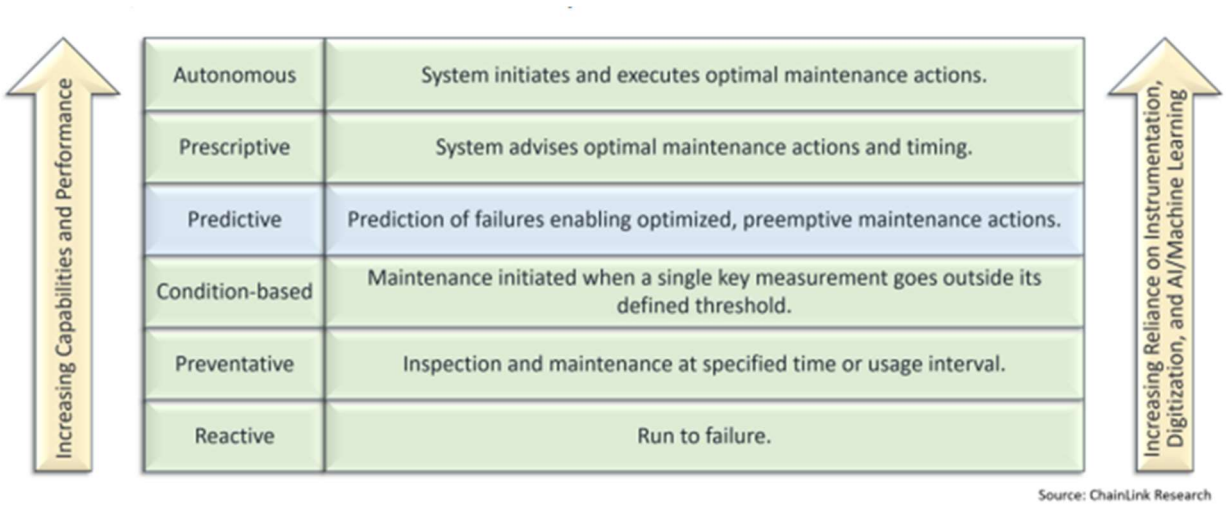


Figure 1. Maintenance maturity model. Image is taken from (Oracle, 2020)

In terms of maintenance management maturity it all start with the run-to-failure mode which is followed by the preventative mode of maintenance which considers the specified time interval between services.

The condition-based maintenance starts to address uptime and maintenance costs by monitoring one or more key measurements, such as temperature, vibration, pressure, or other indicators of an out-of-spec conditions. Thus, maintenance tasks are more likely to be performed when they

are actually needed for this specific machine. Such an approach helps to optimize the maintenance costs as less preliminary services will be conducted and less unexpected failures will happen. However, condition monitoring typically involves monitoring only a few key measurements in isolation, lacking the view of overall asset health and more subtle indicators of deteriorating operation.

Predictive maintenance (PdM) typically involves a broader set of input data and more sophisticated analysis (e.g. motor current analysis, oil analysis, infrared thermography, ultrasonic analysis, etc.). What is more important the predictive maintenance analyses these multiple variables together to provide a more reliable and complex indicator to warn about the service demanded.

The next step of the maintenance management system development is the prescriptive maintenance which takes into account all the data collected in the course of the predictive maintenance, observes various corrective actions taken by maintenance personnel and the outcomes that resulted. Using machine learning techniques, prescriptive maintenance system learns and recommends the best timing and course of action for a given set of conditions.

The final goal of maintenance system development should be autonomous system which initiates and executes maintenance activities itself. (Oracle, 2020)

The topic of the predictive maintenance is well-discussed, there are many prognostic methods that have been presented and discussed in academic papers. 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 (Leohold, et al., 2021), which is clearly shown below (Figure 2).

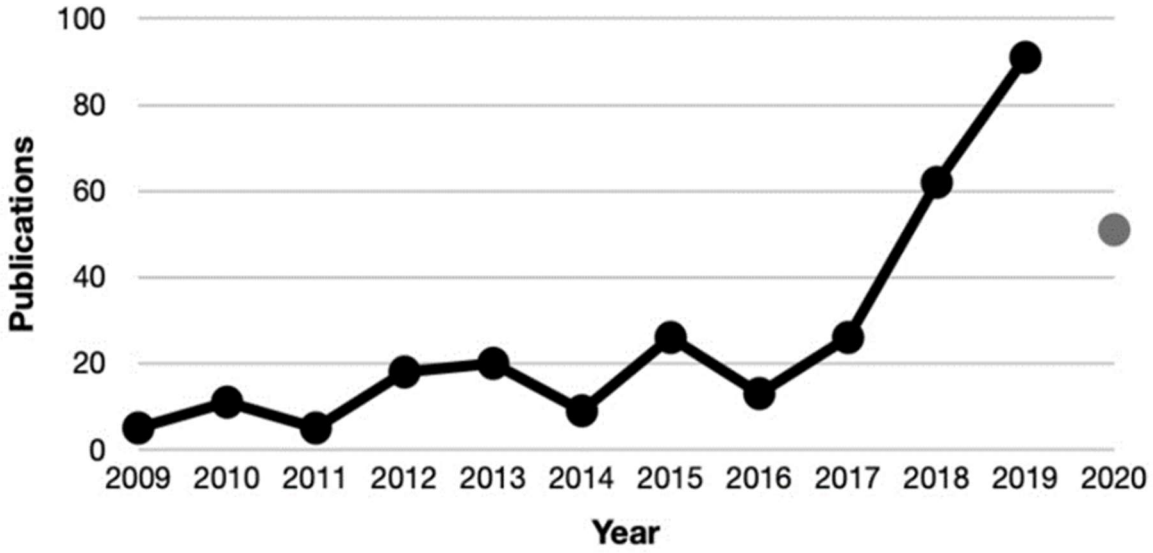


Figure 2. Exponential growth of number of publications considering predictive maintenance. Image is taken from (Leohold, et al., 2021)

Although correct and mature implementation of the preventive maintenance can bring great results to the company there are some shortcuts that can evaporate the technology's potential. According to the McKinsey report (McKinsey, 2019) predictive maintenance can bring poor results for the chemical industry because of the following reasons:

1. Too little data. Unplanned downtime is typically concentrated in a small number of large events so there're usually too few datapoints for PdM systems to learn from

2. Too little time. ML models often predict over time horizons that are too short to be useful in this type of industry where the supply time of certain equipment can be up to a few months.

3. Too little impact. Chemical plants usually operate with a high degree of redundancy which means that if the pump stops unexpectedly, it's always possible to switch to the other line

4. Too little savings. Finally, a focus on reducing unplanned downtime ignores the largest source of throughput losses in most plants. Shutdowns for planned maintenance events cause losses of 5 to 10 percent on average, twice as much as unplanned stoppages.

To avoid these shortcuts the planned maintenance management system should be chosen taking into account all the specific features of the industry, all the management practices described in academic papers or articles should be carefully tailored to needs and peculiarities of the company. Aspects that should be considered while implementing the planned maintenance approaches will be further discussed in 2.3.

2.2. Approaches to RUL estimation

One of the concerns of preventive maintenance is remaining useful life estimation, which is used for 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. Similar classification can be found in many academic articles. The first one considers decisions based on previous failures, while physical models try to model the process. The most accurate predictions in recent research are made through machine-learning based models.

However, several scholars stated that well-known models often lack interpretability of results, and their final metrics highly depend on data quality. Comparison of models considering time-series data was not a focus for many works. In-depth analysis 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.

More detailed information about different types of models is presented below (Figure 3).

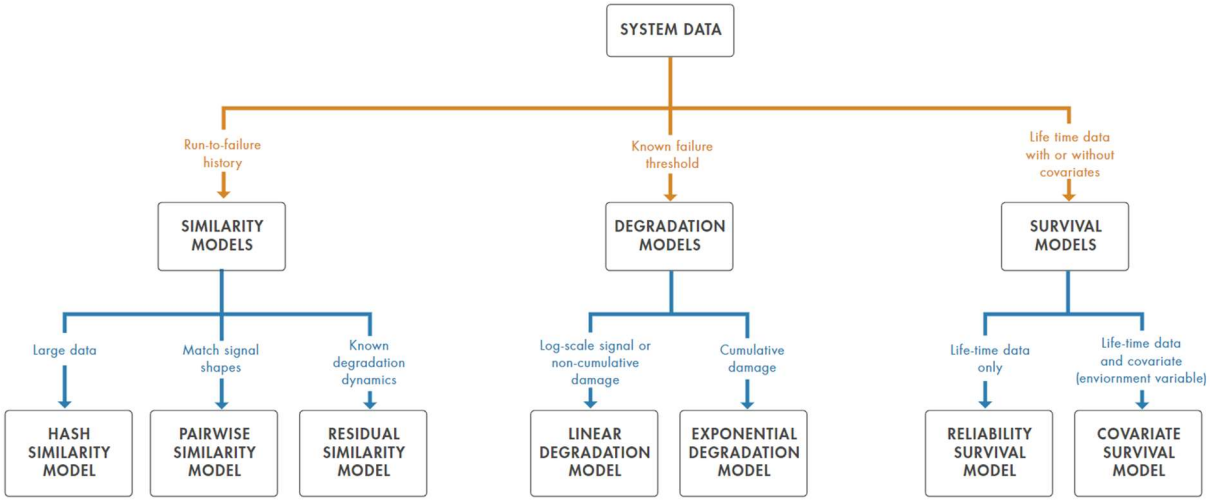


Figure 3. RUL estimation models based on the data available. Image is taken from (Mathworks, 2018).

2.3 Maintenance management practices

Planned machinery maintenance requires proper management techniques to support and develop corresponding business and technological processes. According to (Levitt, 2011), there are some managerial challenges that the maintenance system faces.

The first challenge to be taken into account is the economic issue. One of the main questions that should be answered is if the maintenance routine is worth doing. In other words the cost of the service task and technical support of the maintenance system must be lower than the cost of the consequences of the failure.

According to the (Oracle, 2020), for asset-intensive organisations, the maturity of their maintenance practices is a key determinant of their ability to operate reliably, without interruption, profitably. In the same time it's important to distinguish various types of technical systems and machinery that require different maintenance strategies. Even as the company progresses and adopts each next higher level of maintenance maturity, it should not necessarily abandon less sophisticated maintenance strategies. For example, some companies that implement RCM recognize that a one-size-fits-all maintenance strategy wastes scarce maintenance resources on less critical assets while undeserving more critical assets. The company may even continue to use the run-to-failure approach to very-low-critical and cheap items (e.g. light bulbs, printers, scanners, etc.) ignoring them until they fail. Preventative maintenance may be appropriate for low-medium criticality assets requiring periodic inspection and service, replacement of lubricants, and so forth. However, with preventative maintenance, some

equipment will be serviced before necessary, whereas other equipment will fail before being serviced. (Oracle, 2021)

The company makes the decision about the budget and set of actions based on the 3 levels of economic analysis:

1. macroeconomic analysis. The firm determines whether PM approaches make sense given the organisation's overall goals and the needs and requirements of the business. The most important indicator is the ROI (return on investment).

2. semi-microeconomic analysis. It is considered what strategy is the most appropriate for a particular machine or group of machines being used similarly. Even if a decision has been made at the corporate or plant level to use PM/PdM as the dominant strategy, each machine or machine group has factors that influence how to apply it specifically. Usually, the most important factor is the cost of having the unit out of service (downtime cost). (Levitt, 2011)

The difference in operating time between usual operation and operation with predictive maintenance can be traced on the figure below (Figure 4).

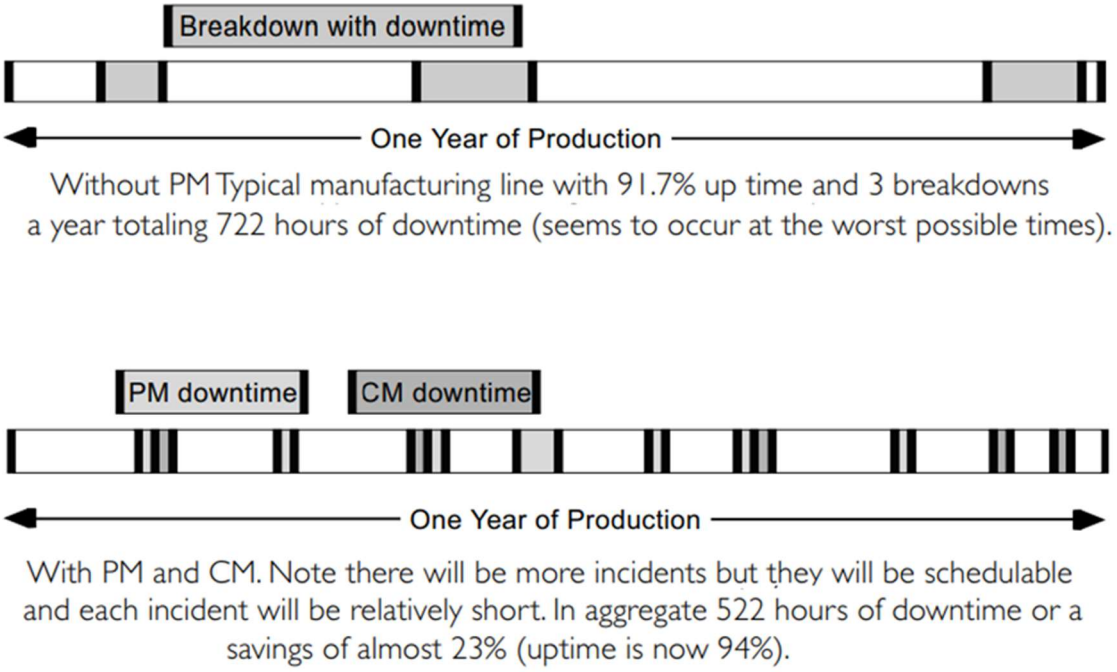


Figure 4. Comparison of downtimes with and without predictive maintenance system. Image is taken from (Levitt, 2011)

3. Microeconomic Analysis (Task Level). At this level the decision is made about strategy for an asset or an asset group. The company must decide what specific PM tasks should be performed for each equipment. In this task view or micro view, the cost and consequence of each task is compared with the cost and consequence of the failure mode the task is trying to avoid.

The economic effect of the PM system implementation can also be expressed as the total maintenance costs calculated as sum of PM costs and breakdown costs. Budget and set of PM operations should be chosen to keep the total maintenance costs within the area of lowest overall costs (Figure 5).

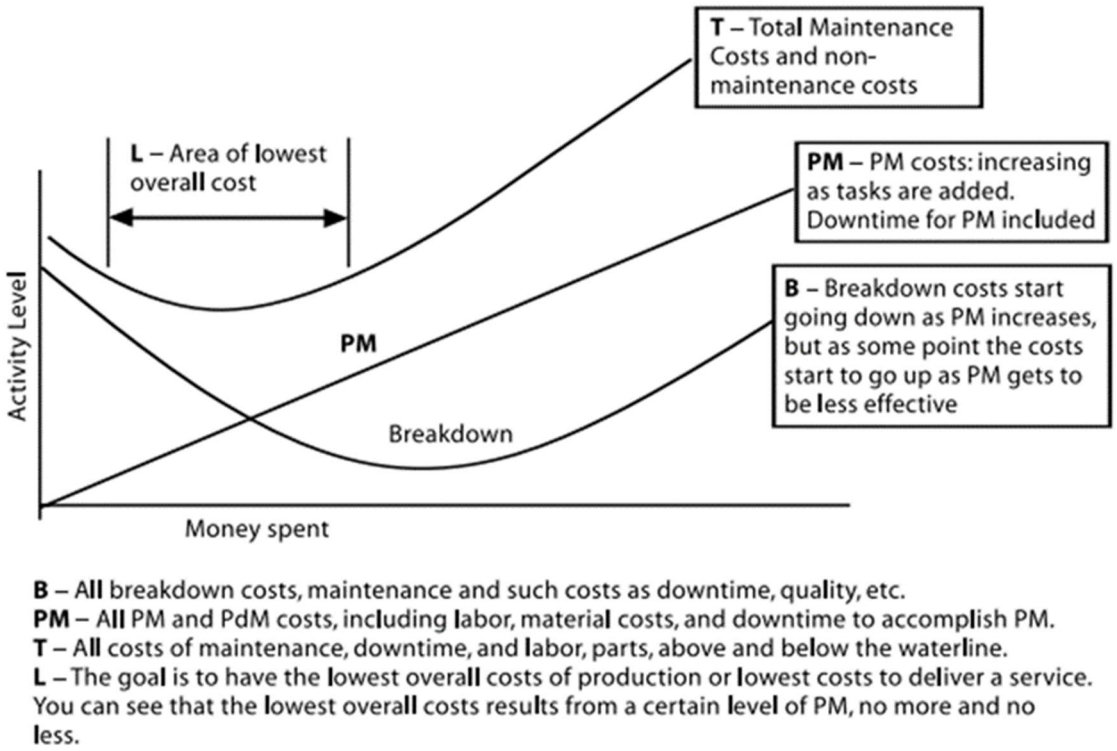


Figure 5. Total maintenance costs curve. Image is taken from (Levitt, 2011)

The 2nd managerial issue is the human factor. People doing the predictive maintenance routine have to be motivated, educated and trained to the extent that they actually do the designated tasks properly. Another thing here is a psychological question that people tend to go for the smaller sure victory and gamble on the larger loss. From the viewpoint of predictive maintenance, it

means that the natural tendency is to agree on small but predictable costs (operate without the predictive maintenance) and to avoid spending money on the predictive maintenance system which will save some money with some probability. The last challenge could be especially difficult at the stage of implementing the planned management system. (Levitt, 2011)

The third managerial challenge is that the planned maintenance system has to be built into the systems and procedures that control the business and these systems must be well- designed to fulfil the business and technological processes required. Information collected has to be integrated into the flow of business information. Predictive maintenance data has to be reported to the Plant Manager or Director of Operations and that should be done in a timely manner.

Predictive maintenance actions should be performed when the equipment is not needed so it takes coordination of business and technological processes planning to achieve the main targets of predictive maintenance - to reduce equipment downtime, reduce costs and improve business efficiency. To be effective, the PM system should be a part of the company's operation and quality management strategy. The foundation for this approach is building and maintaining a 'digital thread' for each asset—i.e. a full-lifecycle, digitally-connected approach to asset management, connecting all of the data and systems for each asset, from concept/design to manufacturing, service, and ultimately end-of-life/recycle. (Levitt, 2011)

Predictive maintenance integrated approach requires aligning technical and managerial IT tools and services. Examples of such a system can be shown on figure 6.

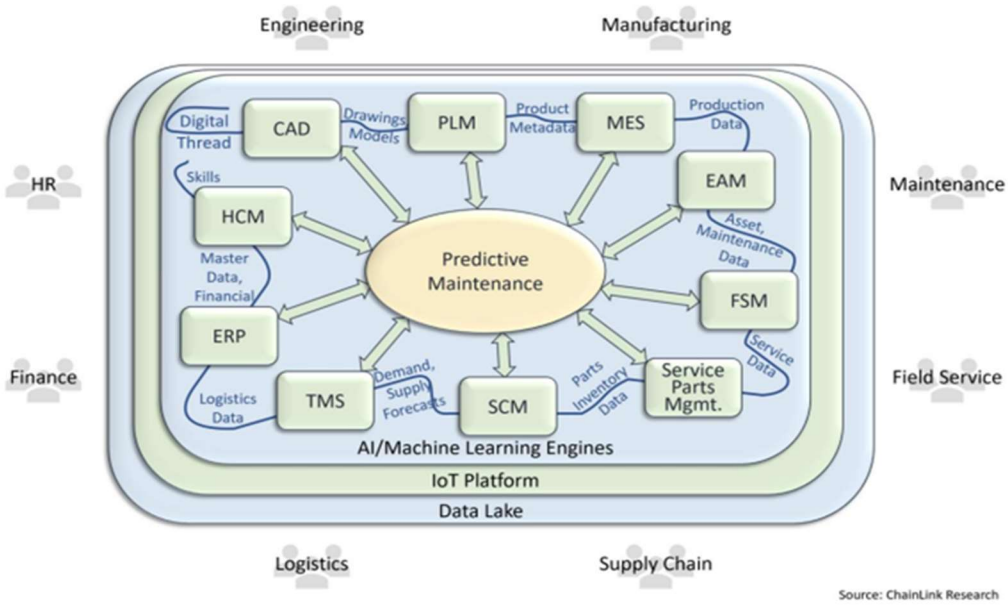


Figure 6. Integrated predictive maintenance schema. Image is taken from (Oracle, 2020)

Management of the PM can be realised with the use of CMMS which is an integrated system that helps the maintenance leadership manage all aspects of life in the department.

The core of a CMMS is its database. It has a data model that organises information about the assets a maintenance organisation is charged with maintaining, as well as the equipment, materials and other resources to do so. (IBM)

The information in a CMMS database supports various functions of the system, which enable the following capabilities:

1. Resource and labour management: Track available employees and equipment certifications. Assign specific tasks and assemble crews. Organise shifts and manage pay rates.

2. Asset registry: Store, access and share various asset information

3. Work order management: Typically viewed as the main function of CMMS, work order management includes information such as: Work order number, Description and priority, Order type (repair, replace, scheduled), Cause and remedy codes, Personnel assigned, and materials used.

4. Preventive maintenance: Automate work order initiation based on time, usage or triggered events.

5. Materials and inventory management: Inventory, distribute and reclaim maintenance and repair operation equipment and materials across storage areas, distribution centers and facilities. Manage suppliers, track inventory costs, and automate resupply.

6. Reporting, analysis and auditing: Generate reports across maintenance categories such as asset availability, materials usage, labour and material costs, supplier assessments and more. Analyse information to understand asset availability, performance trends, MRO inventory optimization and other information to support business decisions and gather and organise information for audits.

CMMS solutions handle functional challenges in a number of ways, but one of the key technological approaches is to deploy them hosted on the cloud as SaaS, in which software is hosted centrally by a vendor and available on demand. (IBM).

3.Company description

3.1. Pharmaceutical companies' specifics of preventive maintenance implementation

In recent years the pharmaceutical industry has experienced major changes in terms of decision-making process, management decisions, data analysis and customer demand. For sure, this industry is one of the most growing sectors with sales of more than 1228.45 billion dollars in 2020. The annual rate of the pharmaceutical market is about 6%. In 2021 worldwide revenue was more than 1462 billion dollars. (Mikulic, 2021)

Unexpected failures may cost companies a huge share of the revenue, so pharmaceutical firms look for any approaches to optimise operating costs and increase profitability. Consequently, there is a strong uptrend in the number of predictive maintenance solutions. For example, North America PdM healthcare market has increased by almost three times from 2017 till 2022.

During the COVID-19 pandemic pharmaceutical companies faced new challenges, so digitised technology is getting more and more important, because it improves performance and creates more accurate planning and forecasting.

The development of the Industry 4.0 had a great impact on the pharmaceutical industry as it bring to existence the Pharma 4.0 which can be defined as the digitalization of pharmaceutical industries from the supply, production (with planning), and delivery operations' points of view by networked firms and uses intensively digital models and ontologies. The related ecosystem includes the process of collecting big data from several sources into a data pool and then performing the necessary analytics using AI, ML, or cognitive process to predict streamline drug life cycle processes. Although Pharma 4.0 is mostly related to the drug development and manufacturing processes, for example, using of AI to design and identify new molecules based on target properties, the broad understanding of the concept includes the following tasks:

- improving the smartness of the contributing resources (i.e., humans, devices).
- connecting and integrating all the contributing resources at any stage of the cycle.
- providing real-time status and awareness information to the regulatory organization.(R. E. Hariry et al, 2021).

Some researchers (V. Steinwandter et al, 2019) mention that the pharmaceutical industry has some specific features which influence the implementation of preventive maintenance and other Industry 4.0 technologies. First of industrial companies experience an increasing need for competitive organised processes as their current main source of income (patent-protected products) run out. This leads to high competition with generic products. Another booster is the Quality by Design concept introduced by the US Food and Drug Administration demanding data- and risk-based approach for the development and manufacturing of drugs. Also the development of new technologies in the pharmaceutical industry faces specific both technical and non-technical obstacles such as:

1. Data access - non-stardatised interfaces and restrictions due to IP protection.
2. Development - data science tools are not integrated into the ecosystem, conservative mindset, and long update cycles.
3. Deployment - software environments were not designed to integrate with data science tools, revalidation of the whole process is needed.
4. Knowledge -algorithms should be tailored to the needs of the company, it's often impossible to reuse available solutions. Also, people who have a specific set of skills are needed to invent, test and deploy these algorithms. Another problem is that many algorithms were developed in the academic environment which lacks knowledge about the real business features. (V. Steinwandter et al, 2019).

3.2. Company overview

BIOCAD is one of the largest biotechnology companies in Russia. 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 – from the search for molecules to mass production - in its own research centres and laboratories.

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. Therefore, BIOCAD tries to automate various processes that are usually carried out manually, for example, filling out documentation and passports for drugs, accounting for laboratory objects.

In the production BIOCAD employers use different devices to conduct analysis and experiments. One of the devices is chromatograph. There are more than 100 chromatograms in production, they have various components and different frequency 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 may also be replaced if this is detected by the researcher. Cost of each lamp is 1200 EUR, and the lamp limit is about 3000 hours. The process is not automatic, which leads to termination of work, because time to order and to deliver the lamp used to take from 2 to 4 weeks and currently this time period could increase up to few months. 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 results of the research are not significant so that the whole process should be repeated, if replacement is conducted between the analysis, it takes only 30 minutes, and no results are lost.

According to historical data the average time between 2 lamps replacement equals to 211 days which means that for 100 chromatographs there will be needed approximately 170 lamp replacements per year. So contribution of the predicted maintenance of 170 processes of lamp replacements per year will save 595 hours of employers working time and 170 preserved research results. In terms of economic efficiency it would mean:

- saving 328 thousand rubles per year on personnel salary (taking into account monthly salary 60 000 rubles for technicians);
- saving at least 1,7 million rubles on chemical reagents which are not lost in case of predictive maintenance.

At the same time the initial implementation of the predictive maintenance system would cost the company around 40 thousand rubles. So the project considered is economically efficient.

Moreover, since the average number of lamps used per year is 20 and the price of the lamp is 1200-euro, lamps delivery costs 24 000 euro and on-time maintenance and lack of necessity to

deliver lamps more than actually needed will have a huge economic impact on the company's budget.

3.3 Company's data pipeline description

For the chromatograph data collection and analysis, the BIOCAD company have created the architecture which is represented on the figure 7.

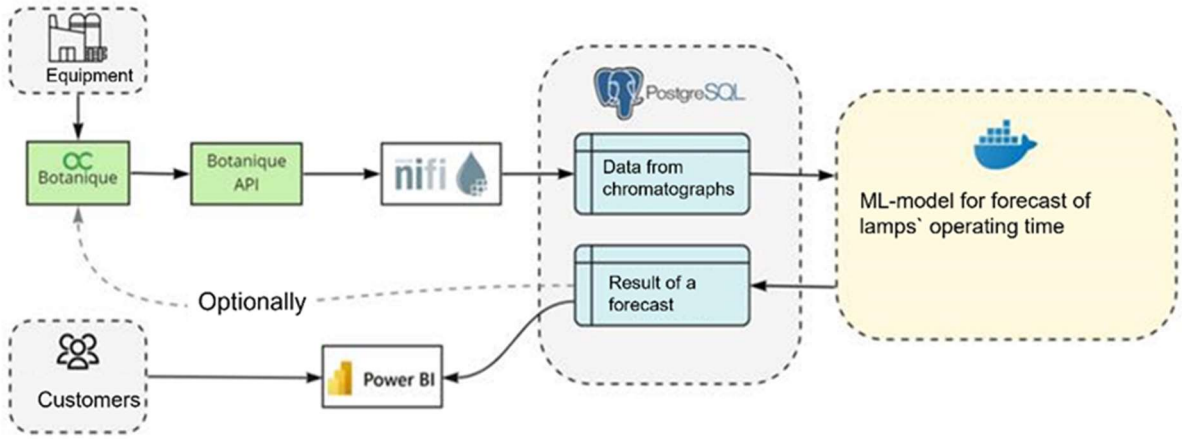


Figure 7. Architecture of the data pipeline. Image is adapted from (BIOCAD, 2022).

The presented architecture includes the following software:

1. Botanique is the software developed by BIOCAD specialists. It allows to collect data from the equipment, fix all the operating processes of the equipment and all the personnel activity, trace climatic conditions of laboratory rooms, form equipment usage reports and dashboards. The software is integrated with other IT systems of the company. Botanique consists of the web-application and telegram chat-bot. Data extraction from the Botanique system is available through API.

2. Apache NiFi is a software designed to automate the flow of data between software systems. It supports powerful and scalable directed graphs of data routing, transformation, and system mediation logic. It has Web-based user interface, it supports data provenance and it's highly configurable, secure and designed for extension. (Apache)

3. PostgreSQL - open-source object-relational database system. Among specific features of the system the following can be mentioned: various data types` support, data integrity assurance, performance, high reliability, security, extensibility and internationalisation. (PostgreSQL)

4. PowerBI is a collection of software services, apps, and connectors that allows to visualise data extracted from different sources Power BI) The PowerBI software brought the Microsoft to the leading position in the 2022 Gartner Magic Quadrant for Analytics and Business Intelligence Platforms (Figure 8).

Source: Gartner (March 2022)

Figure 8. Gartner Magic Quadrant for Analytics and Business Intelligence Platforms. Image is taken from PowerBI official website.

4. Research description

4.1. Python Programming language

Mentioned 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. It allows us to conduct data preprocessing, data analysis, train machine learning models and visualise the results. Mentioned steps will be implemented with the help of the following Python libraries (Table 4).

Table 4. Python libraries used in the thesis.

Task	Python package
Mathematical operations	NumPy
Exploratory data analysis	Pandas
Visualisation of the data	Matplotlib
Metrics for evaluation of calculations	Sklearn
Regression model building	Prophet
Extracting text messages	Re
Processing of the datetime data type	datetime

Source: [authors research]

4.2. Machine Learning

Machine learning application can be used in different spheres – retail, banking, telecommunications, insurance, molecular biology, and genetics. ML 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. Supervised ML algorithms use predetermined labels in addition to the use of input variables. These models try to classify or to predict the output attribute while considering different performance measures.

Time-series analysis is conducted in the thesis with the help of an additive regression model implemented in the Prophet package. The model observes the relationship between the dependent variable (working hours, in this case) and time.

4.3. Remaining useful life estimation

The methods used to calculate RUL depend on the kind of the data available. If there is lifetime data indicating how long it took for similar machines to reach failure, then survival models can be used. If there are run-to-failure histories of machines similar to the one being diagnosed, then similarity models can be used. If there is a known threshold value of a condition indicator that detects failure, then degradation models can be used. (Mathworks, 2018).

In the case presented by BIOCAD, there is a threshold for the lamp usage (3000 hours). However, the lamp can be replaced earlier, moreover, there are no functions or already existing models that can be used for the calculation based on BIOCAD data due to specificity of the data provided, an implementation of a new function or approach is needed.

5. Application of Machine Learning concepts to BIOCAD data

5.1 Input data

The data for the thesis is provided by BIOCAD. 10 datasets from 10 different chromatograms have been used, this dataset concatenated consist of 157 885 rows total. Dataset consists of three columns with information about chromatography activity (Table 5).

Table 5. Example of the initial data.

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>

Source: [authors research]

In sum, the data frame consists of the information from 10 chromatographs. The 6th and the 7th chromatographs` data contribute more to the number of observations than other chromatographs.

'Date' column represents consecutive datetime values of equipment logs. The range of the datetime values are from the 2nd of March of 2015 to the 24th of March of 2022. 'event_class' column provides characteristics of 'text_message'. There are 11 unique values of 'text_message' column with 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

The third column – 'text_message' – consists of 2861 unique values. However, it is important to mention that the 'text_message' column is presented by two parts – detector and textual information, so data extraction is needed to evaluate the 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. Column has such text messages as 'Lamp on', 'Lamp off', so that operating time of lamps can be calculated and measured, also the column has information about the maintenance ('UV-lamp replacement' messages).

Number of text messages which include information about switching of UV-lamps on or off were calculated with the function implemented in Python. Text information was transformed to lowercase format, and rows consisting of 'uv' and 'on', 'uv' and 'off' were counted as the number of 'UV-Lamp on' or number of 'UV-Lamp off' messages respectively. The results are presented below (Table 6).

Table 6. Number of 'UV-Lamp on' and 'UV-Lamp off' messages.

Chromatograph	Number of 'UV-Lamp on' messages	Number of 'UV-Lamp off' messages
0	508	260

1	773	209
2	320	323
3	179	75
4	221	162
5	650	404
6	95	56
7	79	57
8	191	110
9	72	38

Source: [authors research]

As can be seen from the table, all chromatographs present the results where the number of ‘lamp on’ and ‘lamp off’ messages do not coincide. For example, the difference for the zeroth, first and fifth chromatographs is equal to 320, 564 and 246 respectively. It is clear that the number of ‘lamp on’ messages is greater than the number of ‘lamp off’ messages in most cases. It happens because ‘Lamp on’ message can be added to the logs even when the lamp did not ignite, therefore calculation of operating time of lamps requires extraction of pair rows of actual lamp ignition and switching lamp off.

Moreover, ‘text_message’ column provides the information considering the success of the analysis on 10 chromatographs. When a user manually completes an analysis or measurement for any reason, a message ‘aborted by user’ occurs. When the analysis was conducted successfully, there is a message ‘completed’. Numbers were calculated based on text messages including ‘aborted’ and ‘completed’ strings (Table 7).

Table 7. Number of ‘completed’ and ‘aborted’ messages.

Chromatograph	Number of ‘completed ’messages	Number of ‘aborted’ messages
0	722	64
1	384	12
2	338	10
3	325	0
4	262	59
5	1049	1023
6	546	154
7	74	18
8	312	188
9	288	11

Source: [authors research]

Percentage of aborted analyses in the total number of analyses completed was calculated and visualised with the matplotlib library (Figure 9).

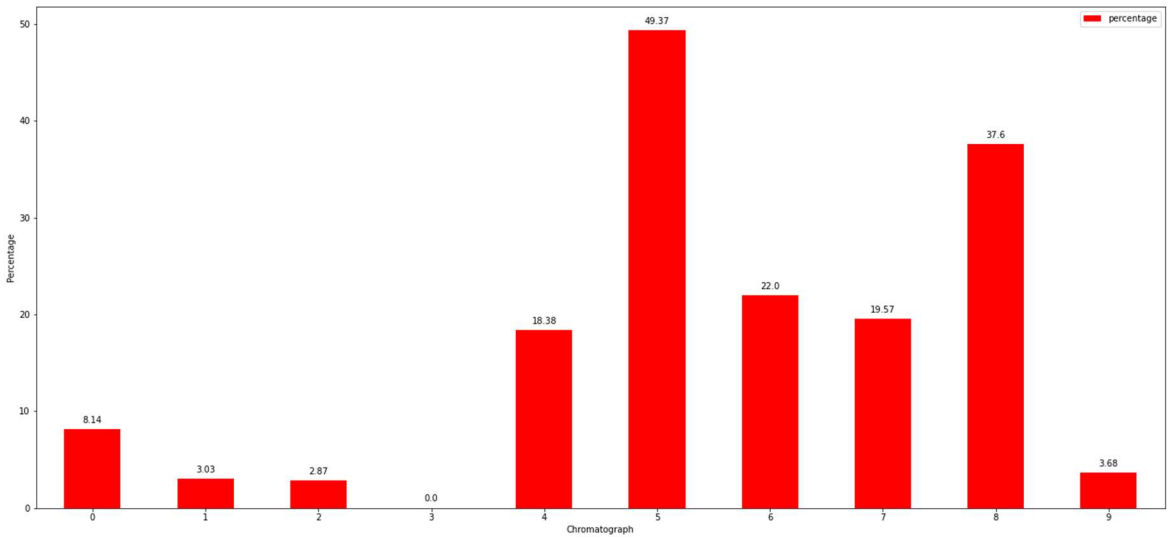


Figure 9. Percentage of the aborted analyses. Source: [authors research]

As can be seen from the figure, there are two chromatographs which presented a high percentage score of aborted analyses - almost 0.5 and 0.4 for the fifth and eighth chromatographs respectively.

BIOCAD also provided us with the information considering actual lamps`operating time for December 2021 and March 2022. Results look as follows (Table 8):

Table 8. Actual working hours of 10 chromatographs.

Chromatograph	Actual working hours (23 Dec 2021)	Actual working hours (24 March 2021)
0	2582	28
1	1626,7	2855
2	1771,5	748
3	2266,7	702
4	1054,7	1678
5	244,7	953
6	495,83	555
7	1561,39	2305
8	The function of viewing working hours is not available	The function of viewing working hours is not available
9	The function of viewing working hours is not available	The function of viewing working hours is not available

Source: [authors research]

Information in datasets is dated 2016-2022 but maintenance reports are added only for 2020-2021 (including information of lamp replacement procedures).

The quality of data represents challenges to deal during the project:

- small number of lamp replacement cases to study due to the fact that this info is available only for 2020-2022.

- information about the amount of working hours is available only for 8 chromatographs and at two points of time only (23/12/2021 and 24/03/2022) .

- data doesn't contain all the timestamps when the lamp was turned on or turned off thus it requires preprocessing.

- non-uniform activity of chromatographs which corresponds to an uneven distribution of time periods between lamp replacements (figure 10). The average time between lamp replacement was calculated to be 197,5 working hours.

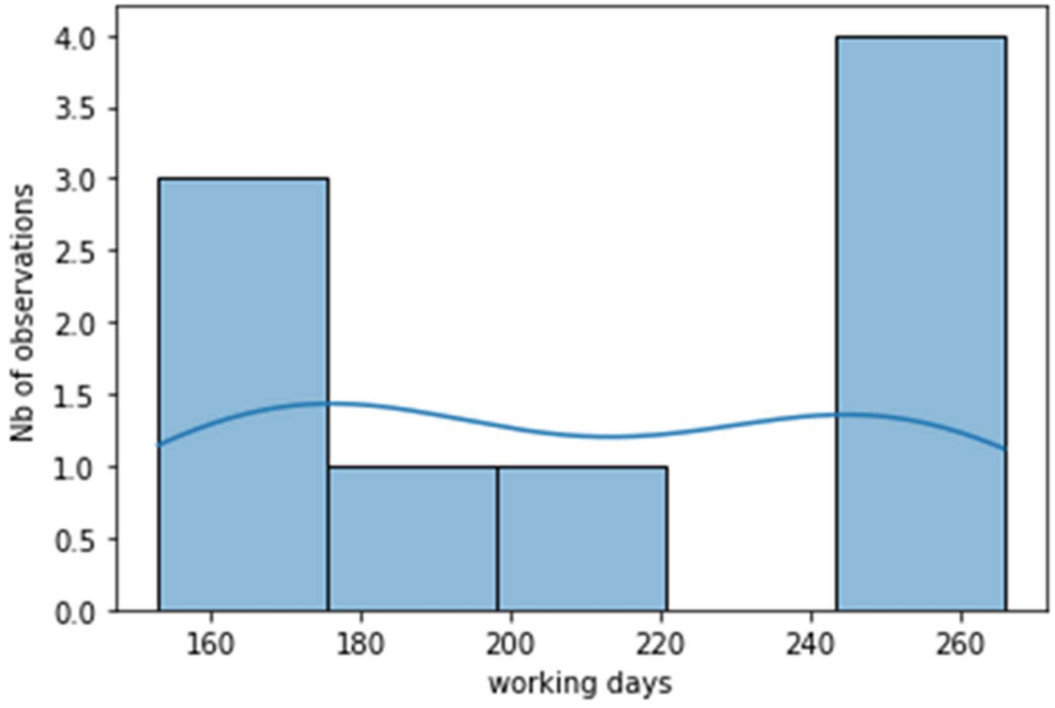


Figure 10. Non-uniform distribution of number of working days per lamp between replacements. Source: [authors research]

For further analysis data obtained has required the following preprocessing:

1. “date” column: change data type to datetime;
2. “text message” column: transforming text messages to lower case and change data type to string.

In the course of the exploratory data analysis all the datasets were divided to subsets which correspond to time periods between lamp replacements or between points with known amount of working hours. The timestamps of lamp replacement were found using Python regular expressions by looking for text messages containing phrases “замена уф-лампы”.

For every subset the following analysis was performed:

1) calculation of messages that signals about the activity of chromatographs (‘get system ready’, ‘system shutdown’, ‘sequence started’, ‘sequence aborted’, ‘sequence completed’), also amount of messages about errors. Results of this analysis for 2 subsets are shown on figures below (Figure 11, Figure 12). On the figure 1 we can see that the amount of messages “lamp on” are not equal to the amount of messages “lamp off” even for small subsets. Some subsets don’t contain messages “lamp on” and “lamp off” but the fact that there’s a message “S started” means that the UV lamp was on at the moment of starting the analysis although we don’t have corresponding messages. This fact brought us to the hypothesis that some other messages should be also taken into account while calculating the operating hours of lamps. This approach will be considered later. The full set of results of this analysis is shown in the appendix.

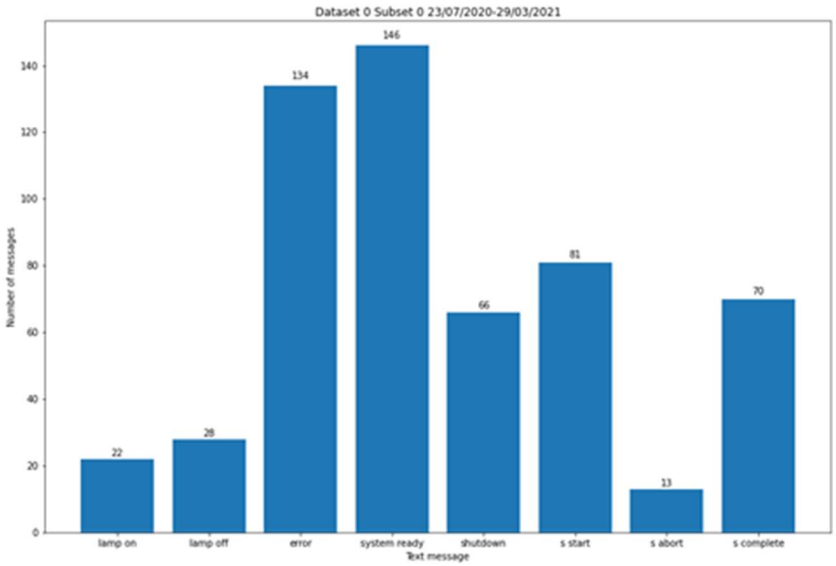


Figure 11. Histogram of messages analysis for dataset 0 subset 0. Source: [authors research]

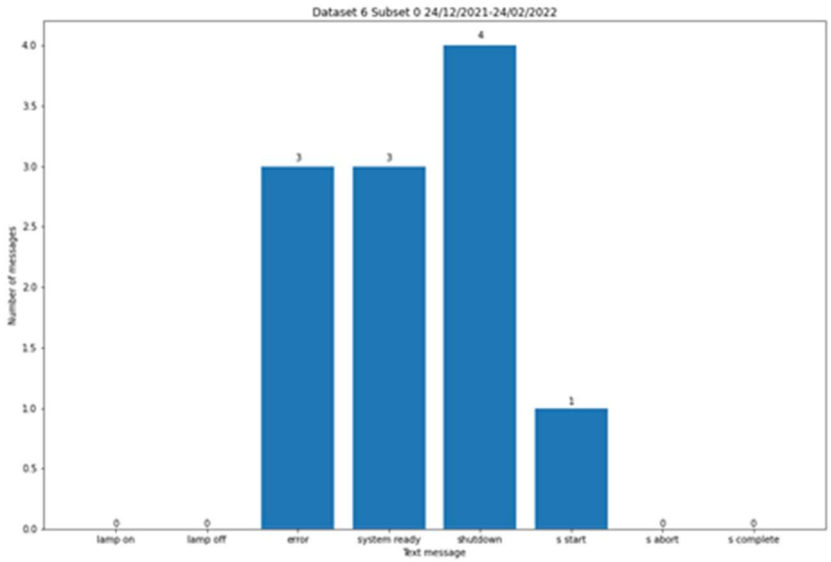


Figure 12. Histogram of messages analysis for dataset 6 subset 0. Source: [authors research]

In the course of this analysis, we have also calculated some indicators which could signalise about communication problems with the certain chromatograph, namely.:

- relation of number messages “lamp on” and “lamp off” (LR);
- relation of number messages “S started” and sum of messages “S completed” and “S aborted” (SR);
- total number of messages consisting “error” which could alert about problems with certain chromatograph (E).

In case of perfect communication with chromatograph the value of first two indicators should be equal to 1. Investigation of subsets showed the results represented in the table 9.

Table 9. Indicators` min and max value.

Indicator	Min value	Max value
LR	0	1.58

SR	0	1.27
E	0	154

Source: [authors research]

Practical application of this analysis will be discussed further.

2) maximal period without any signals from chromatograph. This parameter was calculated as the maximum time difference between the 2 messages in the subset. Results of this investigation are shown in the table 10. As we can see values change significantly for different chromatographs. They may alert about the idle of chromatograph or about communication problems. Practical application of this analysis will be discussed further.

Table 10. Max idle period for each dataset.

№ of Dataset	№ of Subset	Max idle period
0	0	30 days 03:16:48
	1	13 days 04:48:43
	2	12 days 21:55:18
	3	5 days 02:03:36
1	0	13 days 05:54:51
	1	20 days 20:12:00
2	0	58 days 07:24:05
	1	7 days 09:58:59
	2	41 days 05:59:01
3	0	53 days 07:27:11
	1	43 days 13:21:27

	2	31 days 05:59:52
4	0	13 days 20:41:40
	1	7 days 06:52:03
	2	9 days 19:12:33
	3	54 days 22:30:06
5	0	19 days 06:21:20
	1	0 days 00:00:42
	2	8 days 21:24:40
	3	13 days 01:10:34
6	0	37 days 22:31:06
7	0	19 days 20:43:26

Source: [authors research]

Results of the EDA confirmed that data obtained has a complicated pattern of working and standstill periods and some amount of missing values.

5.2. Calculation of the working hours based on the first approach.

Based on the challenges discovered two approaches for calculation of operating time of the lamps were created. The first one is the calculation of working hours based on 'lamp on' and 'lamp off' indicators. Application of the 1st approach faced a few challenges. As mentioned before in 6.1., a lack of important labels was discovered, for example, several data files did not have an equal number of lamp on and lamp off messages, which is crucial information for proper calculation of amount of working hours. Results of the 1st approach are discussed in 6.3. (Calculation of the working hours based on the first approach).

The second approach includes consideration of other indicators such as ‘get system ready’, ‘system shutdown’, ‘sequence started’, ‘sequence aborted’, ‘sequence completed’ as they can also be ‘start’ and ‘stop’ flags and some other hypotheses which will be discussed in 6.4. (Calculation of the working hours based on the second approach).

For calculating of working hours with the first approach the indexes are used for identification of corresponding ‘lamp on’ and ‘lamp off’ indexes were identified. After that a new dataframe with timestamps of lamp turning on and off and time difference between rows is created. The data frame is used to calculate cumulative sum of working hours using cumsum() function in Python, so that gradual increase in operating time is clearly seen. The function was used to calculate the working hours of all lamps for 10 chromatographs. The resulting data frame looks as follows (Table 11).

Table 11. Data after transformation function implementation.

Index	Dates	Hours	Hours_total
0	2020-09-24	4.3	4.3
1	2020-09-25	0	4.3
2	2020-09-26	0	4.3
3	2020-09-27	0	4.3
4	2020-09-28	0	4.3

Source: [authors research]

As can be seen from the table, cumulative sum is calculated accurately, so that there is information of working hours for each consecutive day. The results were visualised using Matplotlib() library for each separate subset and for the whole historic activity of chromatographs. As we can see from 13 and figure 14 chromatographs can show different operating patterns and future behaviour doesn’t necessarily correspond to historic activity which creates additional challenge to calculate the operating hours and make predictions.

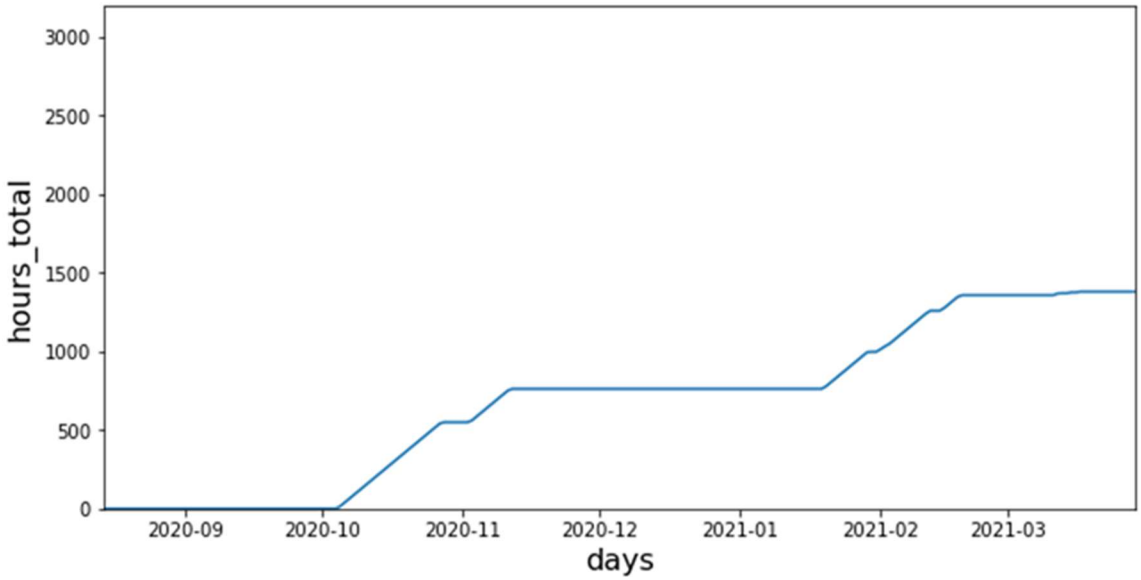


Figure 13. Highly non-uniform graph of the historical activity of chromatographs. Source: [authors research]

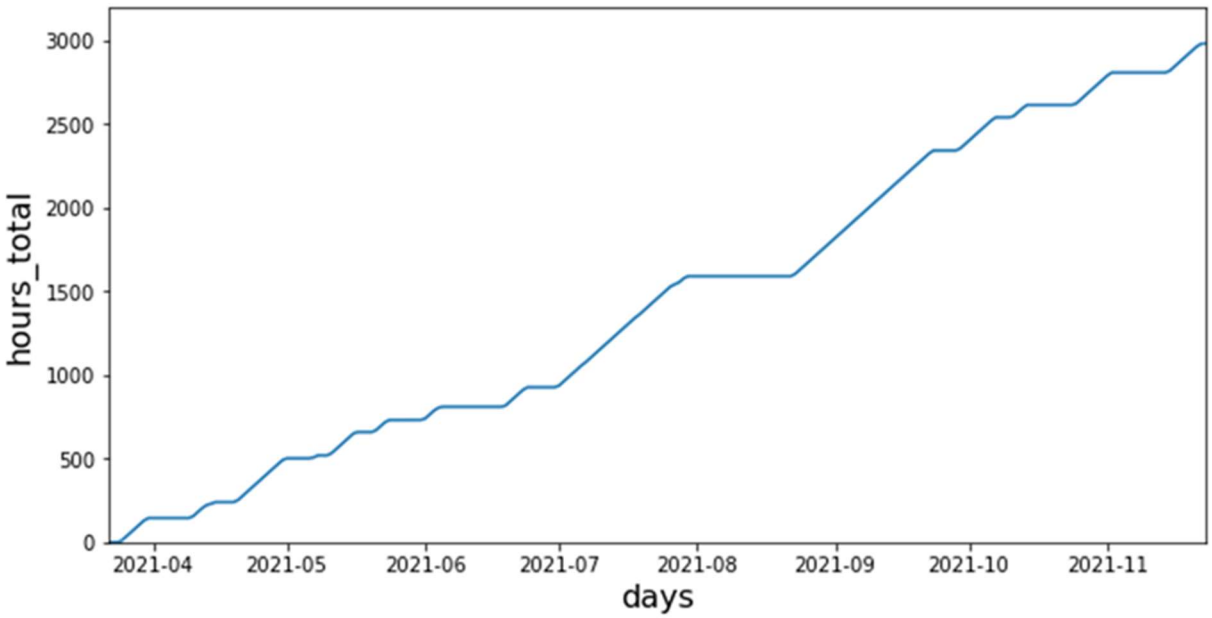


Figure 14. More uniform work of chromatographs. Source: [authors research]

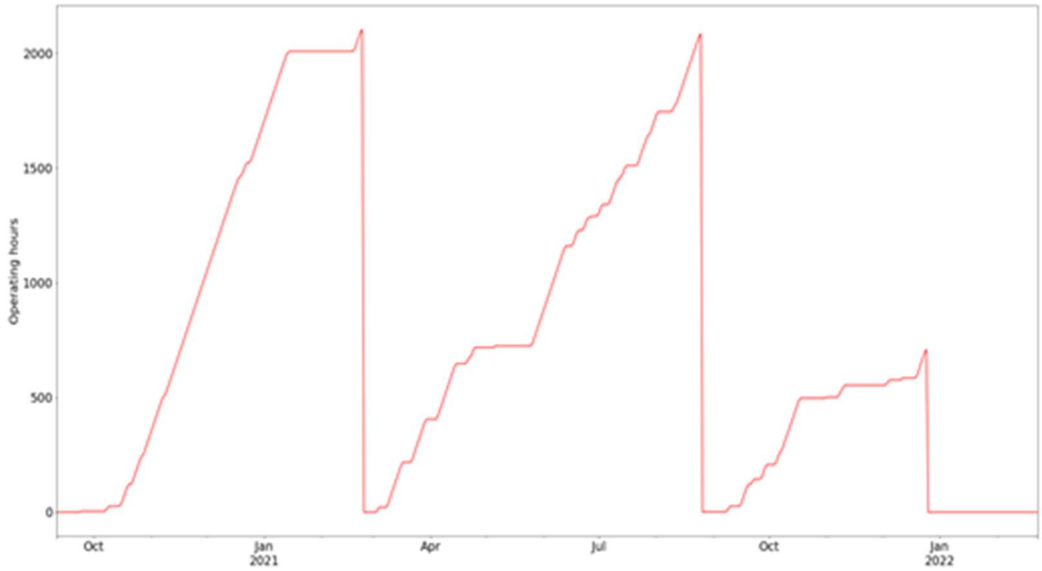


Figure 15. Lamps` operating time of the 4th chromatograph. Source: [authors research]

As can be seen from the figure, three lamps were working on the 4th chromatograph. The graph shows gradual increase of operating time with rapid decrease at the point of lamp`s replacement. As was already mentioned, the usual limit of operating time of the lamp is about 3000 hours, however, in practice, and as the graph confirms, lamps can be replaced at the 2000 hours operating time or even at 700 hours.

Since the BIOCAD has provided us with the actual lamp`s operating time, the calculation error of the working hours calculation based on the first approach was observed. The following function has been used:

$$p = (a - b)/b * 100 \quad (1)$$

where p is a percentage error, a is calculated with the help of the function working hours of the lamp, b is the actual total working hours provided by the company.

The percentage error is presented below (Table 12).

Table 12. Calculation error of the operating time based on the first approach.

Data set number	Predicted lamp`s operating time (h)	Actual lamp`s operating time (h)	Calculation error (%)
0	11,44	28	59,14
1	2235,9	2855	21.69
2	3722,5	748	397,67
3	3150,5	702	348,79
4	609,8	1678	63,66
5	429,5	953	54,93

Source: [authors research]

There are different results for data sets, some of them are irrelevant. For example, there are high calculation errors - for the second, third chromatograph. It can be connected to the problem of the missing data considering lamps` replacements, which affects the calculation error, since the function cannot divide the data into subsets accurately. At the same time, according to BIOCAD's representatives` opinion, there are calculation errors which are acceptable (the zeroth and first data set) .

Then regression models for 15 lamps were built. Operating time was forecasted using an additive regression model from the Prophet() package. Prophet is optimised for the business forecasts tasks with the hourly, daily, or weekly observations with at least a few months, preferably a year, of history. Prophet is able to detect changes in trends by selecting changepoints from the data automatically. (Letham et al., 2017) An example of the forecast graph looks as follows (Figure 16).

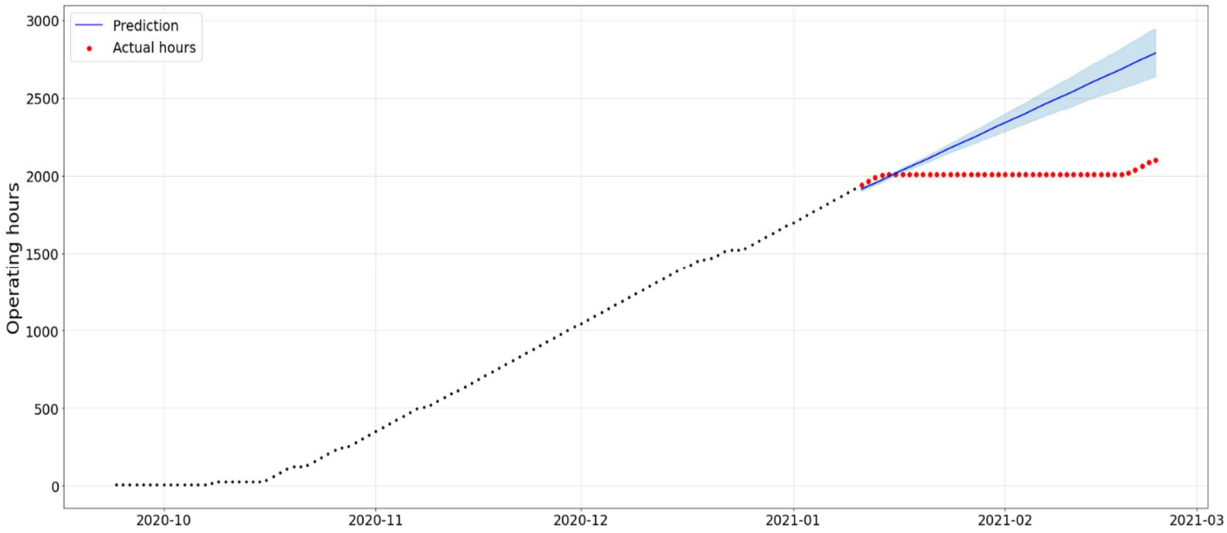


Figure 16. Regression model for prediction of the lamp`s operating time. (Chromatograph 4, lamp 1). Source: [authors research]

Regression models showed comparatively good results for cases when there are no standstill periods and there is a general trend as presented above. However, lamps can be switched off for long periods of time. It is more difficult for the model to make forecasts in this case. The visualisation looks as follows (Figure 17).

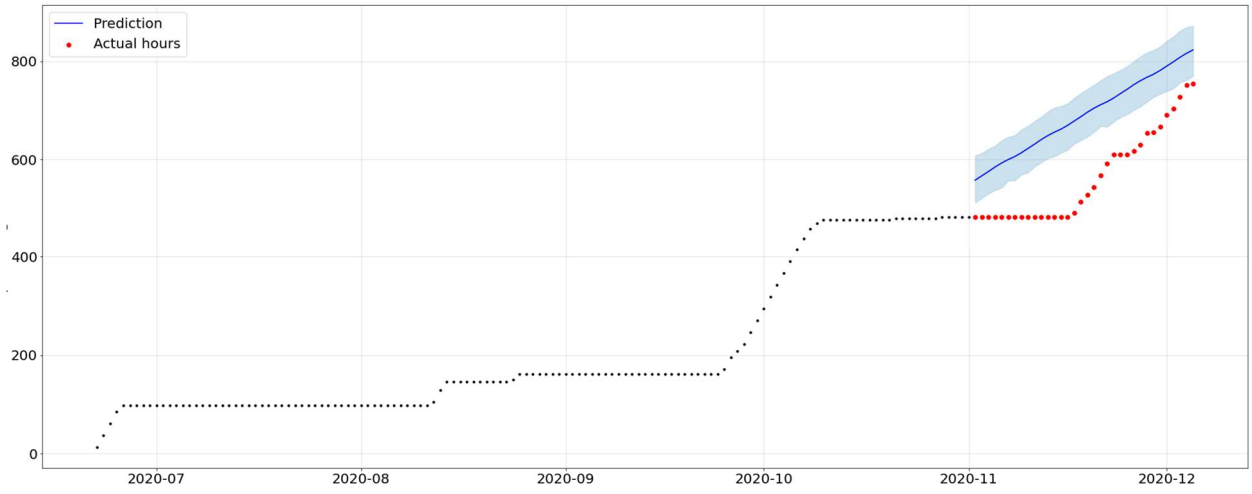


Figure 17. Regression model for prediction of the lamp`s operating time. (Chromatograph 1, lamp 1). Source: [authors research]

All regression models were assessed using Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) metrics. These metrics are interpretable and understandable. MAE

tells how far the forecast's distance from the true value on average, MAPE is an absolute error normalised over the actual value. MAPE captures more errors and outliers. Combination of two metrics was used to assess models` results (Vahdeput, 2019). MAE and MAPE were calculated using the following formulas:

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} \quad (2)$$

where y is the prediction, x is the true value and n is the number of times the summation iteration happens.

$$MAPE = \frac{\sum \frac{|A-F|}{A} * 100}{N} \quad (3)$$

where F is the prediction,A is the true value[EVS1] and N is the number of times the summation iteration happens.

Result for each model`s forecast accuracy looks as follows (Table 13).

Table 13. Regression models forecasts` accuracy.

Data set number	0			1			2		3	
Lamp	1	2	3	1	2	3	1	2	1	2
Mean Absolute Error	108	783,7	Was active for one day	127,5	5,7	117,5	33,1	32	60,8	0,8
Mean Absolute Percentage Error	8	40		23,8	6,9	5,4	3	1	1,9	0

Data set number	4			5			
Lamp	1	2	3	1	2	3	4
Mean Absolute Error	346,7	400,4	26,8	211,7	279,9	Was active for two days	71
Mean Absolute Percentage Error	17,2	22,3	4	10	10		24,1

Source: [authors research]

Two forecast models showed better results than calculations conducted with the help of the function based on the first approach.

All in all, the first approach proved to be accurate within the cases where a full data is provided, however, when there is a problem of missing data, error metrics are high.

5.3. Calculation of the working hours based on the second approach.

Taking into account all the challenges of the data obtained we have developed the following hypothesis to calculate the amount of operating hours more precisely:

1) besides messages “lamp on” and “lamp off” other messages should be considered as flags signalising the beginning (start flags) and the end (stop flags) of lamp’s operation. Here we can consider messages “get system ready trigger” and “S started” as flags of operation start and messages “system shutdown” and “S completed” or “S aborted” as flags of operation stop;

2) all service messages (written in russian language in equipment logs) signalise idle periods of chromatograph;

3) absence of any messages during certain periods of time should be also considered as the standstill of chromatograph (or lost communication);

4) taking into account the 1st or the last message of identical messages going in a row changes the result.

According to hypothesis mentioned above 10 models were developed to calculate working hours of lamp for each chromatograph for all subsets. Description of these models is available in the table below (Table 14).

Table 14. Models` description (Second approach).

Model №	Model description
1	Start flags: lamp on, Get System Ready trigger, S started Stop flags: lamp off, shutdown, S stopped, S aborted Max period of messages absence which is not considered as the idle period 24 hours Two identical flags in a row collapse to the 1st one of them
2	Parameters are the same as of model 1, but max period of messages absence which is not considered as the idle period is 7 days
3	Start flags: lamp on, S started Stop flags: lamp off, S stopped, S aborted Max period of messages absence which is not considered as the idle period 24 hours Two identical flags in a row collapse to the 1st one of them

4	Parameters are the same as of model 3, but max period of messages absence which is not considered as the idle period is 7 days
5	Start flags:lamp on Stop flags: lamp off Max period of messages absence which is not considered as the idle period 24 hours Two identical flags in a row collapse to the 1st one of them
6	Parameters are the same as of model 5, but max period of messages absence which is not considered as the idle period is 7 days
7	Start flags: lamp on Stop flags: lamp off Max period of messages absence which is not considered as the idle period 24 hours Two identical start flags in a row collapse to the 1st one of them, two identical stop flags in a row collapse to the last one of them
8	Parameters are the same as of model 7, but max period of messages absence which is not considered as the idle period is 7 days
9	Start flags:lamp on Stop flags: lamp off Max period of messages absence which is not considered as the idle period 24 hours Two identical flags in a row collapse to the las one of them
10	Parameters are the same as of model 9, but max period of messages absence which is not considered as the idle period is 7 days

Source: [authors research]

Results of calculation of the amount of working hours and errors for all the subsets where actual working hours are known is shown in the appendix.

Having analysed the result of calculation we can come to the following conclusions:

1) Data subsets with good quality of data and less missing data have relatively good results for almost all models considered while other subsets with a lot of missing data show mainly poor results for almost all models.

2) Some subsets show better performance in case of odd models (with max allowed period without messages 24 hours) while other show better performance in case of even models (with max allowed period without messages 7 days).

3) Model 7 shows equal or more working hours calculated than model 5 which for most subsets is closer to the actual value.

For most chromatographs with high values of error the following problems were mostly identified:

- missing data. For example, for chromatograph 1 the last message related to the work of equipment is dated 29/12/2021, after that date we have only service messages. But actual value of working hours for chromatograph 1 for the whole period 24/12/2021 - 24/03/2022 equals to 1228,3 hours which means that some data about operation of chromatograph has been lost;
- missing service messages. For example, for the chromatograph 5 there was probably the additional UV-lamp replacement on 23/11/2021 as the service message says "выполненные работы:замена лампы и оптической ячейки" but without precise information what type of lamp was replaced, UV-lamp or VIS-lamp. The big value of positive error allows us to conclude that at the considered point of time the UV-lamp has been replaced.

Based on the results analysis we have decided to create the 11 model which combines model3, model4 and model7, namely it considers the following hypothesis:

-start flags: "lamp on" and "S started";

- stop flags: "lamp off", "S aborted", "S completed";
- max period of messages absence which is not considered as the idle period is set to be 3 days.
- two identical start flags in a row collapse to the 1st one of them, two identical stop flags in a row collapse to the last one of them.
- in case the amount of working hours calculated exceeds the guaranteed time limit (3000 hours), the warning message is printed and the amount of working hours for this dataset is re-calculated by subtracting 3000 hours from the previously calculated amount of working hours.

So, in case of exceeding 3000 hours of operation the program shows the warning message including information about the time period within which the lamp replacement event should have taken place.

```
Warning! Work hours exceeded the set! (start date: 2021-01-15, end date: 2021-12-23)
```

Figure 18. The warning message.

Result of subsets analysis according to the 11 model is shown in table 15. Results were calculated only for those subsets which have a significant amount of information to be splitted into train and test datasets.

Table 15. Results of the 11th model.

№ of DS	№ of Subset	Time period	Predicted working hours	Actual working hours	Error, %
0	1	29/03/21 - 17/12/21	2112,75	2582	-18,17
1	0	15/06/21 - 23/12/21	2485,8	1626,7	52,81
	1	24/12/21 - 29/12/22	120,21	1228,3	-90,21
2	1	15/01/21 - 23/12/21	1618,49	1771,5	-8,64
3	0	02/10/21 - 15/12/21	456,03	2266,7	-79,88
	2	10/02/22 - 17/03/22	845,8	702	20,48
4	2	26/08/21 - 21/12/21	1085,18	1054,7	2,89
	3	25/12/21 - 24/02/22	23,75	1654,25	-98,56
5	2	22/03/21 - 21/12/21	883,09	244,7	260,89
	3	24/12/21 - 17/03/22	1050,06	708,3	48,25
6	0	24/12/21 - 24/02/22	1,44	59,17	-97,57
7	0	24/12/21 - 17/01/22	29,82	743,61	-95,99

Source: [authors research]

In the course of data analysis and calculation of lamp's working hours also other indicators were extracted, namely:

- average number of hours of non-stop work of the lamp and standard deviation of this parameter;
- average number of hours of the standstill period of the lamp and standard deviation of this parameter.

Results of this investigation for the 13th model are shown in table 16.

Table 16. Working and standstill parameters of lamps for each subset.

№ of DS	№ of Subset	Work non-stop, hours		Idle periods, hours	
		Average	Standard deviation	Average	Standard deviation
0	0	155.2	199.6	42.84	65.58
	1	60.86	121.44	30.18	58.09
	2	72.58	115.59	11.4	26.85
	3	130.27	4.75	32.87	29.54
1	0	53.85	75.03	62.15	73.68
	1	0.13	0	60.11	10.02
2	0	349.32	568.38	81.8	143.47
	1	28.55	49.41	36.37	52.07
	2	0.35	0.46	27.25	21.43
3	0	131.5	94.11	41.46	98.15
	1	0	0	0	0
	2	0	0	845.8	0

№ of DS	№ of Subset	Work non-stop, hours		Idle periods, hours	
		Average	Standard deviation	Average	Standard deviation
4	0	53.87	92.27	36.34	71.83
	1	31.34	49.37	36.89	67.05
	2	55.71	86.81	33.91	42.65
	3	41.96	0	11.88	11.88
5	0	27.32	71.79	35.53	81.31
	1	0	0	0	0
	2	32.53	60.56	46.23	67.68
	3	28.77	67.32	30.88	90.52
6	0	0	0	1.44	0
7	0	271.44	205.29	9.94	10.0

Source: [authors research]

As we can see from table 16 for almost all subsets the value of standard deviation is very high which confirms the fact that activity of chromatographs is highly non-uniform. On the other hand, it may alert communication errors with certain chromatographs due to which the analysis shows longer idle or working periods than that really observed for the chromatographs. Meaning of these values from the maintenance point of view will be discussed in 7 as these parameters may be considered as health indicators for communication with devices, chromatographs themselves and business processes of the company.

Results for regression analysis for the datasets constructed based on model 11 are presented in the table 17.

Table 17. Regression analysis results based on model 11.

№ of DS	№ of Subset	Time period	Predicted working hours	Mean average error, hours	Mean average percentage error, %
0	1	29/03/21 - 17/12/21	1823	108.84	5.56
1	0	15/06/21 - 23/12/21	2405.73	57.00	2.39
2	1	15/01/21 - 23/12/21	4511.98	44.28	0.97
3	0	02/10/21 - 15/12/21	696.66	157.51	34.84
	2	10/02/22 - 17/03/22	857.61	1.61	0.21
4	2	26/08/21 - 21/12/21	926.62	88.24	8.75
5	2	22/03/21 - 21/12/21	3744.82	149.80	3.92
	3	24/12/21 - 17/03/22	1205.03	88.49	8.58
7	0	24/12/21 - 17/01/22	74.90	27.12	90.93

Source: [authors research]

5.4. Implementation of the model

For the successful implementation of the model it should fit the existing IT ecosystem of the company and existing business processes. To fulfil the 1st condition the program developed provides the following information:

- current status of the chromatograph based on the last flag identified (start or stop) which will allow operators to understand which devices are in operation or in standstill state right now and use the equipment for evenly;

- graphs of cumulative sum of operation hours for each chromatograph which will allow operators to see the general trend of chromatographs operation;

- prediction of operation activity in the future to understand whether there's a need for lamp replacement in the next 2 weeks and conduct all the necessary business processes in time;

- mean idle and non-stop working hours for each chromatograph to better understand the operating pattern of different chromatograph and use equipment for evenly and efficiently;

- warning messages that the amount of operating hours calculated have exceeded the guaranteed value of 3000 hours and probably the service message of lamp replacement has been lost. These messages will allow operators to put attention to service messages absence which may lead to big errors of prediction algorithms;

- warning messages if the current amount of idle hours or non-stop working hours is more than the average value. These messages will alert operators about the possible communication problem due to which the current behaviour of chromatograph doesn't fit with the average activity;

- warning messages if the absolute value of difference between 1 and relation of messages "lamp on" and "lamp off" or relation of messages "get system ready trigger" and "system shutdown" or relation of messages "S started" and sum of "S aborted" and "S completed" is more than the preset value. In the program the pre-set value is equal to 0.25, i.e. the alert message will be generated if the relation of messages is lower than 0,75 or more than 1,25.

These messages will alert operators about the possible communication issues due to which part of messages is lost. The important remark here is that equality of number of messages "lamp on"

and “lamp off” doesn’t mean that there’re not any communication problems because messages could follow each other like this:

lamp on
lamp on
lamp off
lamp off

The considered case practically means that at least 1 message “lamp off” and 1 message “lamp on” are lost as the correct consequence should look as follows:

lamp on
lamp off
lamp on
lamp off
lamp on
lamp off

So in this examples number of messages “lamp on” equals the number of messages “lamp off” but communication problem still exists. On the other hand, the fact that relation is far from 1 definitely signalises about communication errors.

Example of the warning message is shown on figure 19.

```
Warning! Idle hours exceeded the average idle hours! (value: 2619.77, avg: 0.13)
Warning! Relation of messages on/off exceeded the allowed value! (value: 0.5, set: 0.25)
Warning! Relation of messages ready/shutdown exceeded the allowed value! (value: 1, set: 0.25)
Warning! Relation of messages S start/S stop exceeded the allowed value! (value: 0.56, set: 0.25)
```

Figure 19. Example of the warning message. Source: [authors research]

Based on all the data generated by the program we have created the dashboard in PowerBI which has all the information available and provides operators with data to make business and technical decisions. The example of such a dashboard is shown on the figure 20.

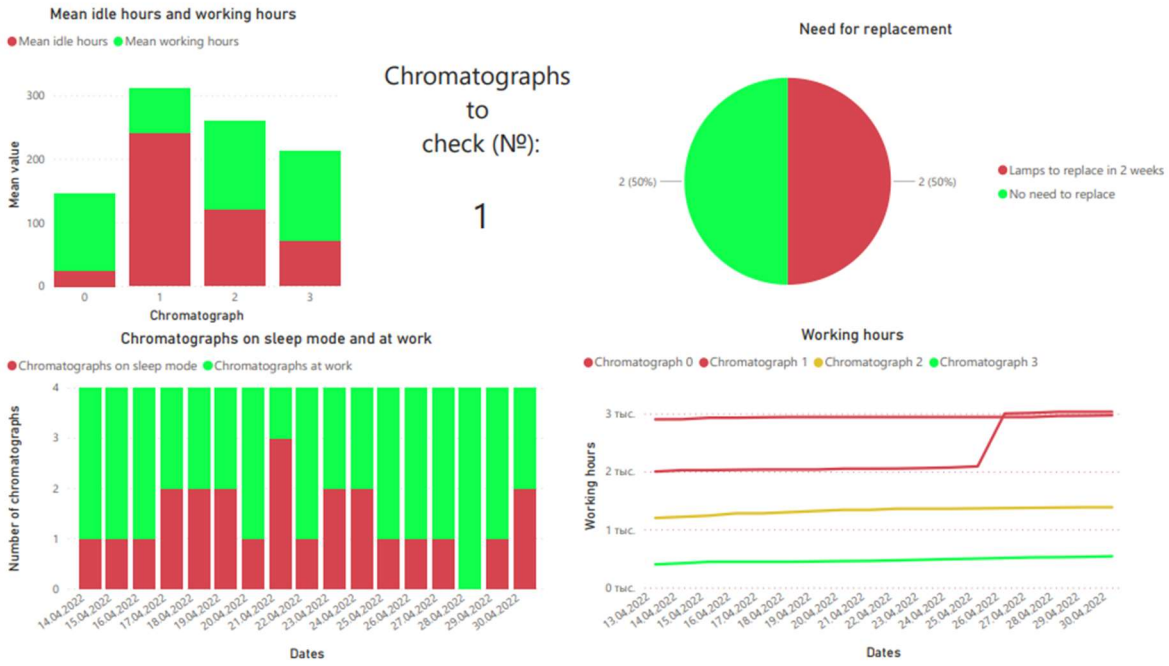


Figure 20. Example of the dashboard. Source: [authors research]

The dashboard allows operators to get all the operational information about chromatographs on day-to-day basis and also provides them with information about the amount of lamps to be replaced soon. The dashboard also contains numbers of chromatographs which data has generated the alert messages due to exceeding of average working or idle hours or due to the relation of the amount of certain text messages which may signal about communication errors with chromatographs.

The most important recommendation for the successful implementation of the model is to find sources of losing the data and fix them as we have understood and proved in the course of the analysis conducted that datasets without significant problems with missing data show relatively good performance for the models considered. By missing values both signals from chromatographs and service messages are meant.

During the model implementation the value of non-stop working hours or idle hours which should generate the alert message should be clarified depending on the real operational activity of chromatographs.

Considering the whole IT ecosystem of the company it can also be recommended that the warning messages are sent to operators using the Telegram bot. This system will allow operators to get to

know about chromatograph potential issues as soon as possible and not to lose important information.

The approach developed can be used also for other categories of equipment which works periodically, has the pre-defined guarantee lifetime and is characterised by the relatively low level of automatisation, for example:

1) medical equipment (for example ultrasound diagnostics equipment) and its parts;

2) industrial equipment of low-medium criticality (for example lubricant pumps) and its parts.

Conclusion

The research conducted has been devoted to the investigation of equipment logs of chromatographs of the BIOCAD company and development of the model for calculation of working hours of UV-lamps and prediction timestamps for replacement of UV-lamps at least 2 weeks before that urgent need arises.

The work has been based on the CRISP methodology and Python capabilities.

In the course of the analysis data has been preprocessed and EDA has been conducted. The main challenge of the data is missing data that make it tough for some datasets to develop model that show good performance. Another problem is the uneven operating activity of chromatographs and lack of any relation between the historic activity and future performance.

As the result of analysis 12 models for working hours calculation have been analysed and 2 of them were chosen for the prediction generation. Results of these two models have also been analysed and described.

Work results have been visualised using the Power BI tool. The dashboard for chromatograph operators has been developed which demonstrates the graph of operating hours for all chromatographs and also general information about historic activity of chromatographs. For the need of further implementation of the model the set of warning messages have been created which are aimed to signal operators about potential problems with communication.

References

An introduction to predictive maintenance [Book] / auth. Mobley R. Keith,. - 2002

Apache NiFi official site.[online] Available at: Apache NiFi(Accessed 16 April 2022)

Ask an expert: What Industry 4.0 can do for maintenance .[online] | Available at: Ask an expert: What Industry 4.0 can do for maintenance | McKinsey (Accessed 16 April 2022)

Forecast KPIs: RMSE, MAE, MAPE&Bias. [online] Available at: https://towardsdatascience.com/forecast-kpi-rmse-mae-mape-bias-cdc5703d242d#:~:text=The%20Mean%20Absolute%20Error%20(MAE,mean%20of%20the%20absolute%20error. (Accessed 10 April 2022)

ISO 13372:2012(en) Condition monitoring and diagnostics of machines.[online] | Available at: ISO 13372:2012(en), Condition monitoring and diagnostics of machines — Vocabulary (Accessed 16 April 2022)

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.

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

Manufacturing: Analytics unleashes productivity and profitability.[online] Available at: Manufacturing: Analytics unleashes productivity and profitability | McKinsey (Accessed 16 April 2022)

Mathworks. Three ways to estimate Remaining Useful Life. [online] Available at: https://www.mathworks.com/content/dam/mathworks/ebook/estimating-remaining-useful-life-ebook.pdf (Accessed: 12 March 2022)

McBeat B.(2020) The Value of Foresight. Generating Value Through Integrated Predictive Maintenance[online] Available at: value-of-predictive-maintenance.pdf (oracle.com) (Accessed 16 April 2022)

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)

PostgreSQL official site.[online] Available at:PostgreSQL: About(Accessed 16 April 2022)

PowerBI official site.[online] Available at:What is Power BI? - Power BI | Microsoft Docs(Accessed 16 April 2022)

Predictive Maintenance: A Novel Framework for a Data-Driven, Semi-Supervised, and Partially Online Prognostic Health Management Application in Industries [News Article] / auth. Francesca Calabrese 1 *, Alberto Regattieri // Applied Science. - 2021.

Predictive maintenance: the wrong solution to the right problem in chemicals.[online] Available at: Predictive maintenance: the wrong solution to the right problem in chemicals | McKinsey (Accessed 16 April 2022)

R. E. Hariry, R.V. Barenji, A. Paradkar (2021) From Industry 4.0 to Pharma 4.0.[online] Available at: (PDF) From Industry 4.0 to Pharma 4.0 (researchgate.net)(Accessed 16 April 2022)

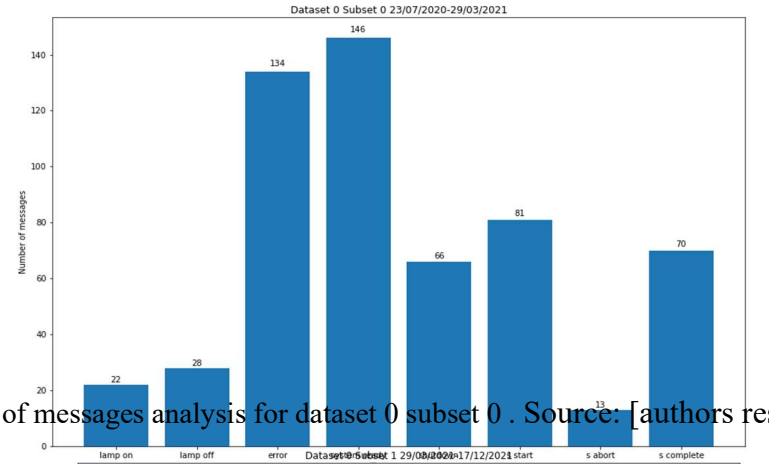
Tao F., Qi Q., Ang L., Kusiak A. (2018) Data-driven smart manufacturing. 48. *Elsevier*. 157–69. doi:10.1016/j.jmsy.2018.01.006.

V. Steinwandter, D. Borchert, C. Herwig (2019)Data science tools and applications on the way to Pharma 4.0.[online] Available at: (PDF) Data science tools and applications on the way to Pharma 4.0 (researchgate.net) (Accessed 16 April 2022)

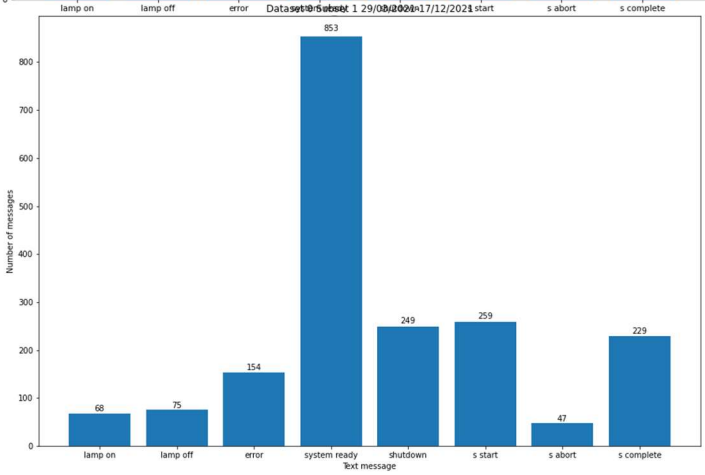
What is preventive maintenance?[online] Available at: What is preventive maintenance? | IBM (Accessed 16 April 2022)

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.

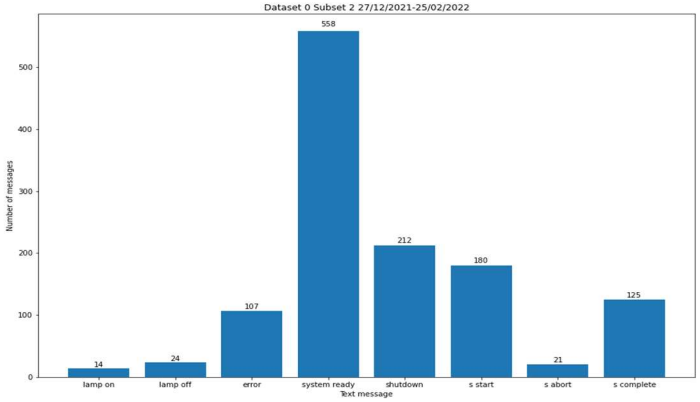
Appendix



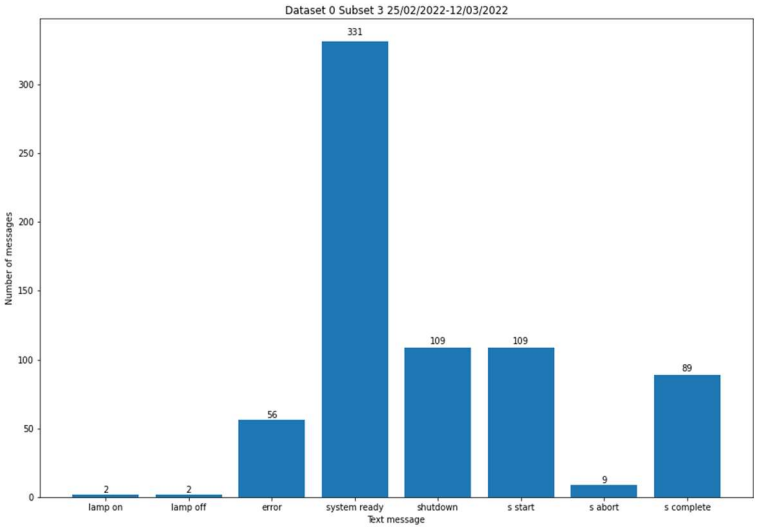
Histogram of messages analysis for dataset 0 subset 0 . Source: [authors research]



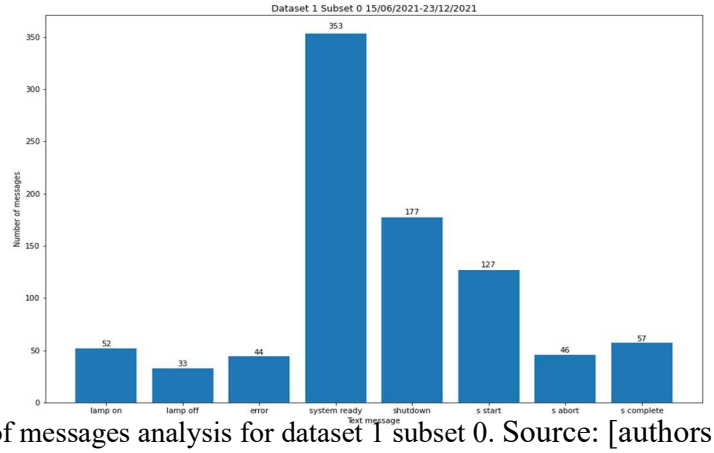
Histogram of messages analysis for dataset 0 subset 1. Source: [authors research]



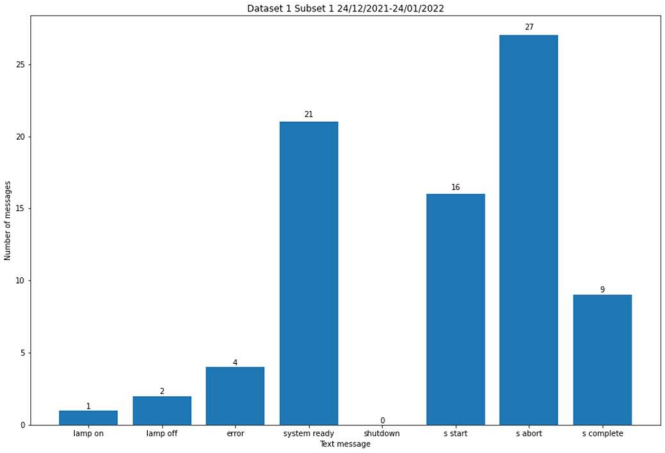
Histogram of messages analysis for dataset 0 subset 2. Source: [authors research]



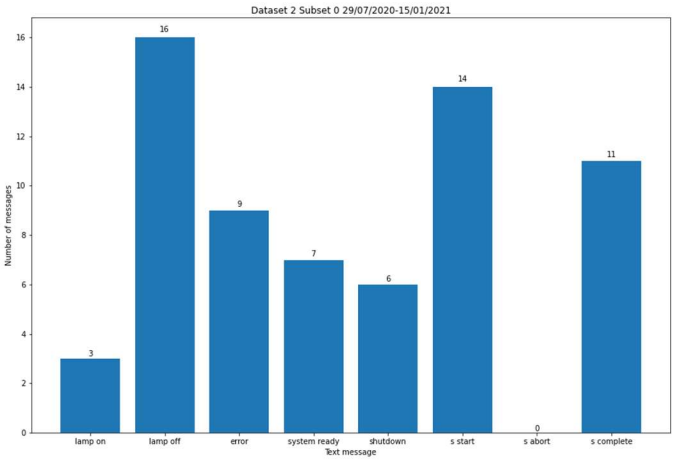
Histogram of messages analysis for dataset 0 subset 3. Source: [authors research]



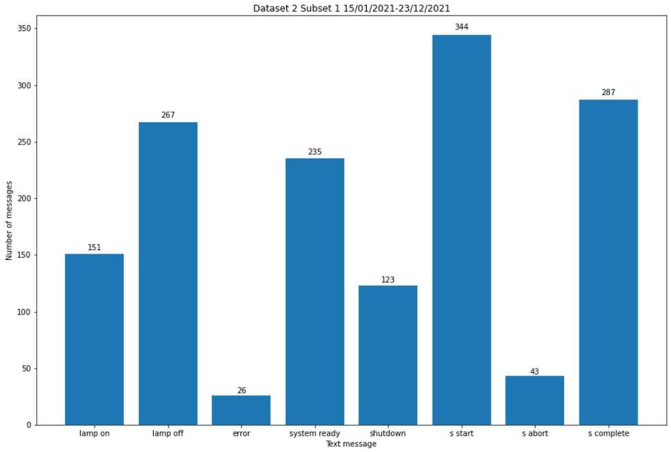
Histogram of messages analysis for dataset 1 subset 0. Source: [authors research]



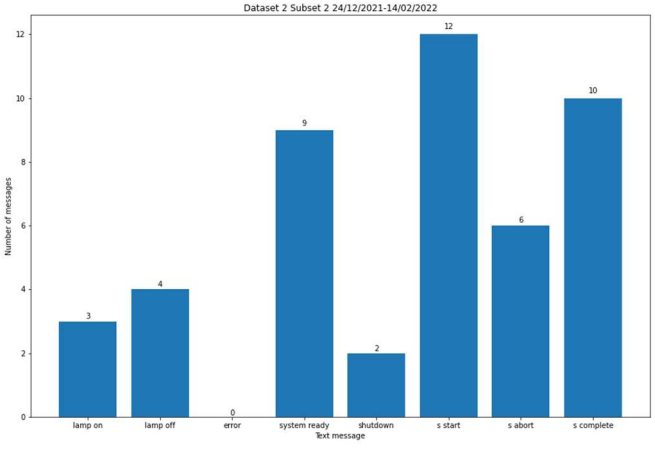
Histogram of messages analysis for dataset 1 subset 1. Source: [authors research]



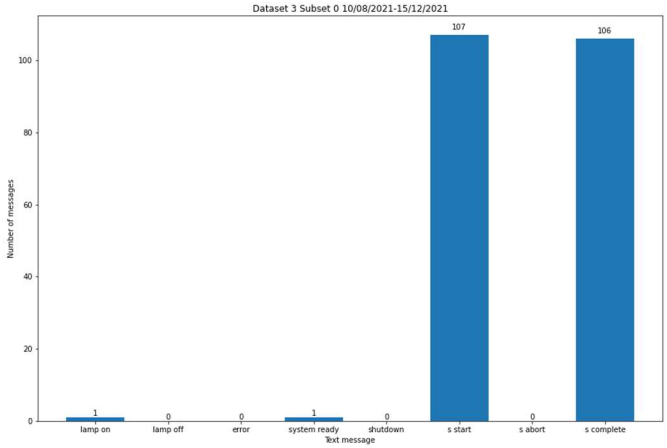
Histogram of messages analysis for dataset 2 subset 0. Source: [authors research]



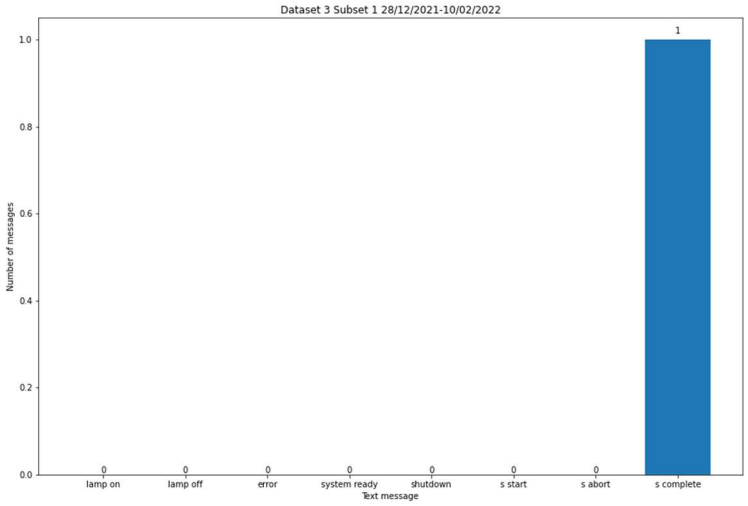
Histogram of messages analysis for dataset 2 subset 1. Source: [authors research]



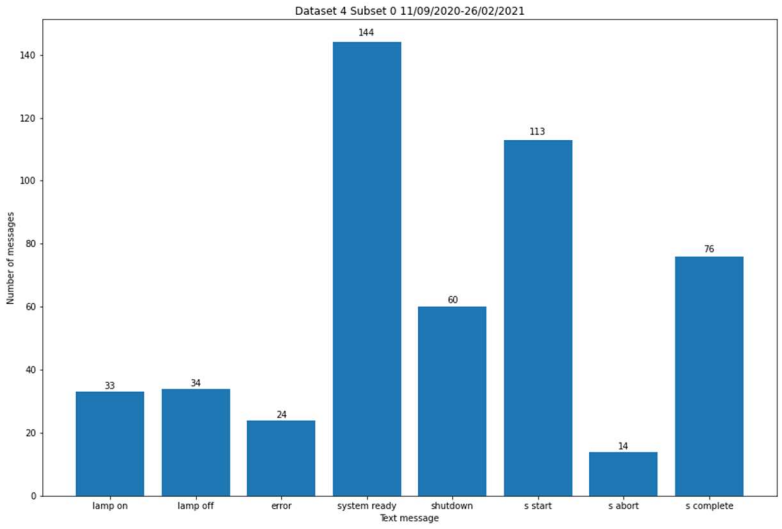
Histogram of messages analysis for dataset 2 subset 2. Source: [authors research]



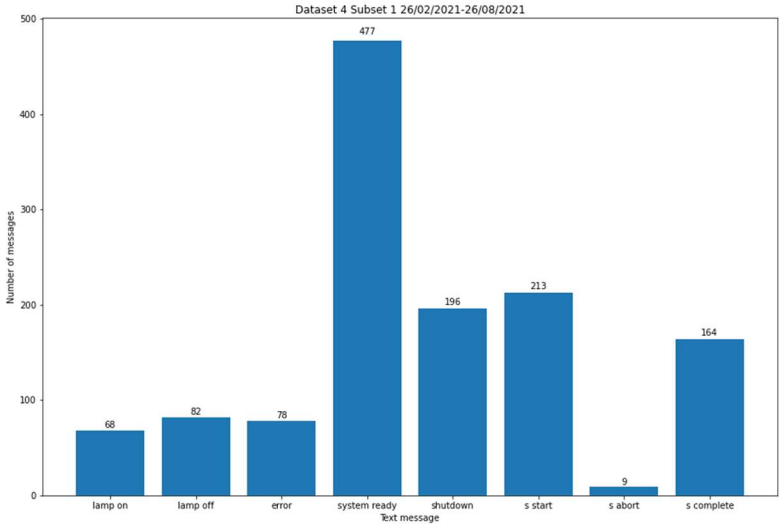
Histogram of messages analysis for dataset 3 subset 0. Source: [authors research]



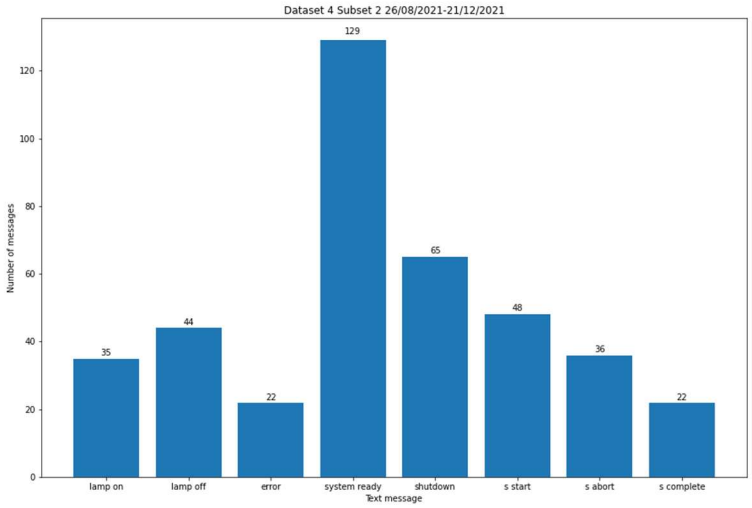
Histogram of messages analysis for dataset 3 subset 1. Source: [authors research]



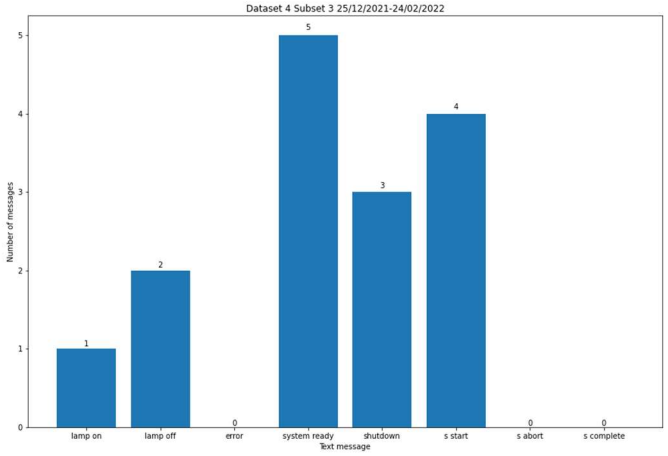
Histogram of messages analysis for dataset 4 subset 0. Source: [authors research]



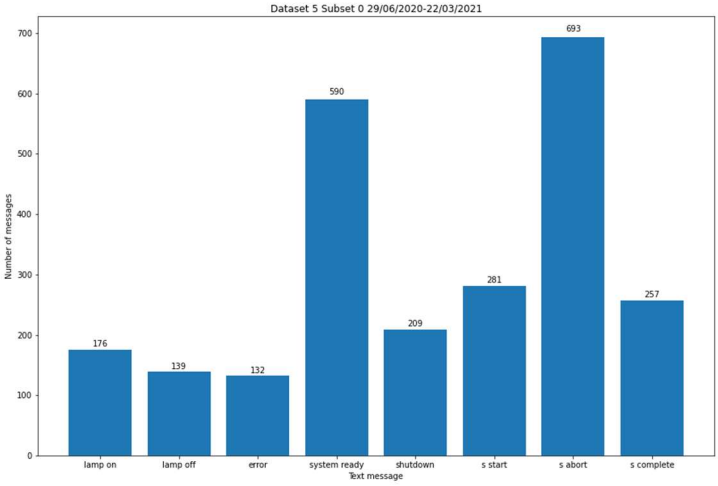
Histogram of messages analysis for dataset 4 subset 1. Source: [authors research]



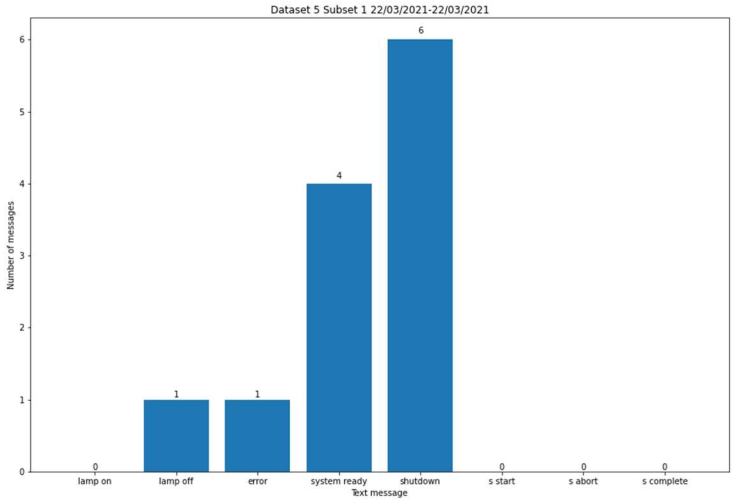
Histogram of messages analysis for dataset 4 subset 2. Source: [authors research]



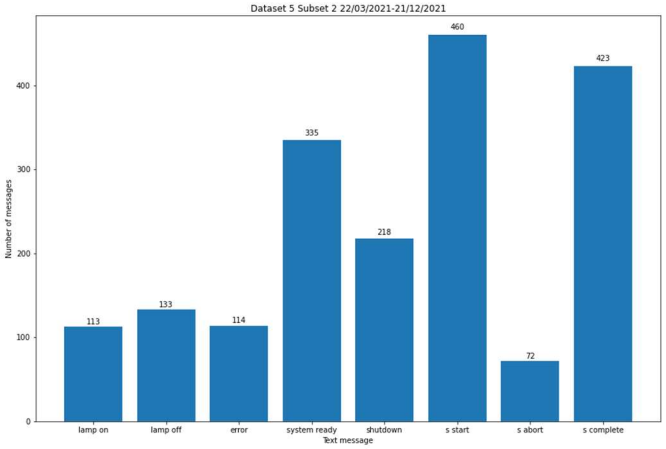
Histogram of messages analysis for dataset 4 subset 3. Source: [authors research]



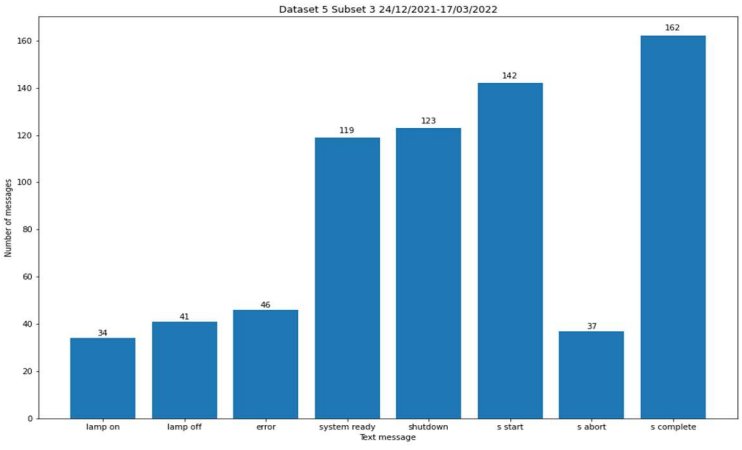
Histogram of messages analysis for dataset 5 subset 0. Source: [authors research]



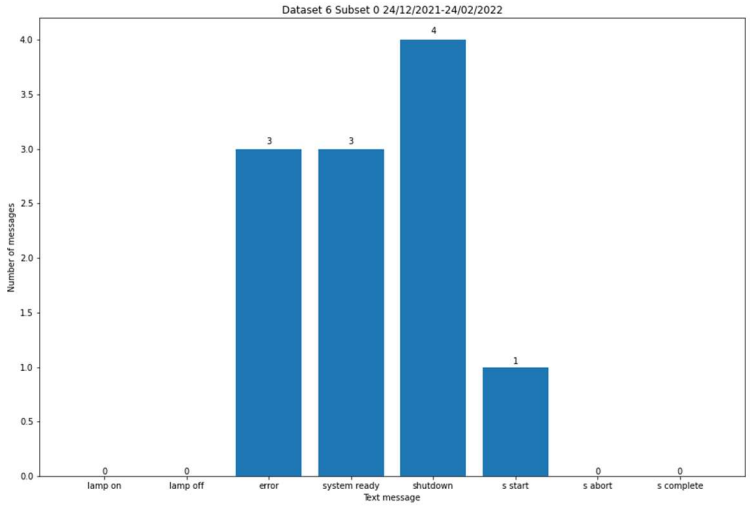
Histogram of messages analysis for dataset 5 subset 1. Source: [authors research]



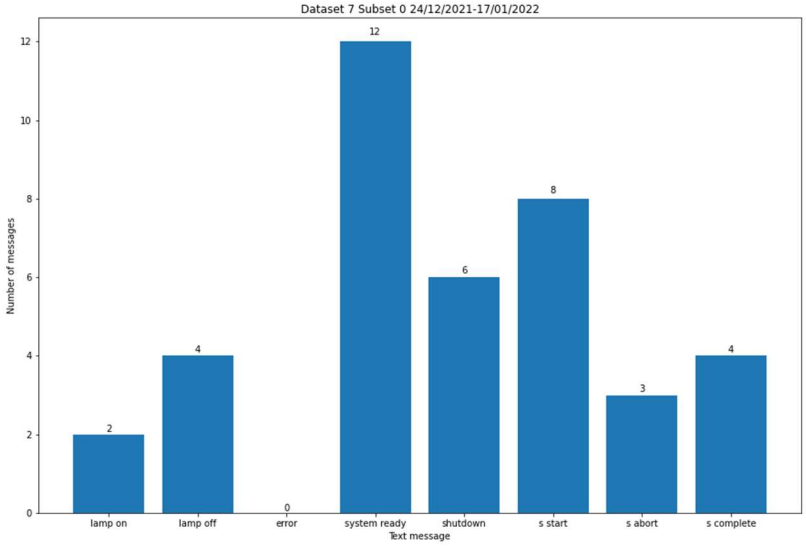
Histogram of messages analysis for dataset 5 subset 2. Source: [authors research]



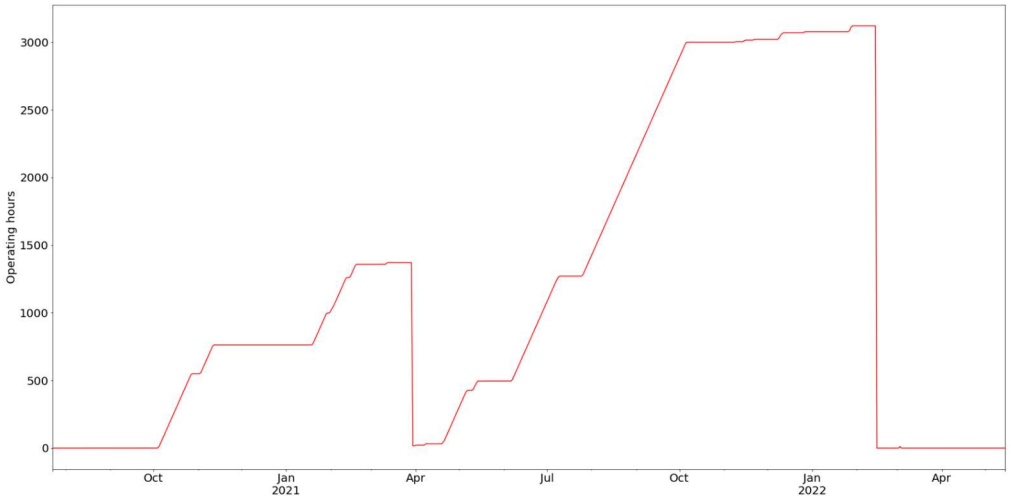
Histogram of messages analysis for dataset 5 subset 3. Source: [authors research]



Histogram of messages analysis for dataset 6 subset 0. Source: [authors research]

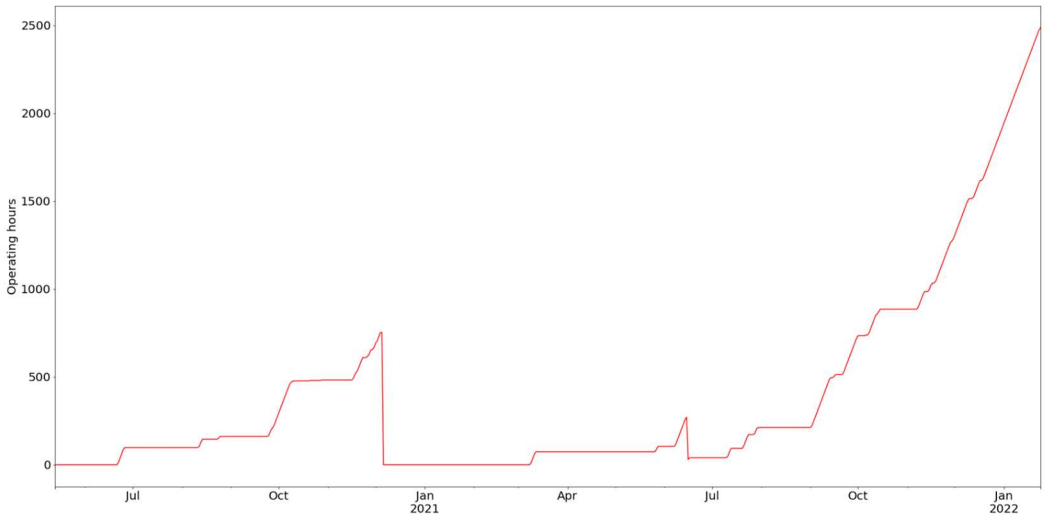


Histogram of messages analysis for dataset 7 subset 0. Source: [authors research]



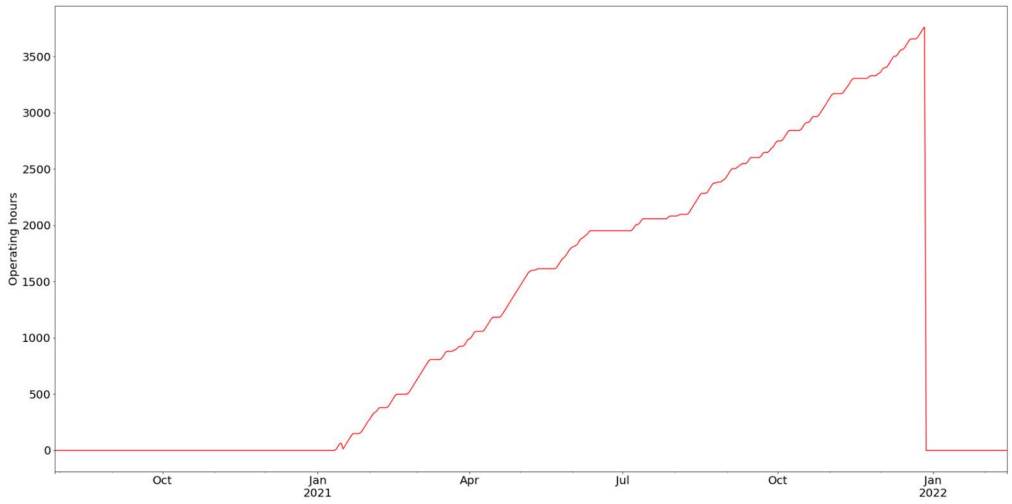
Calculated lamps` operating time of the 0th chromatograph (First approach).

Source: [authors research]



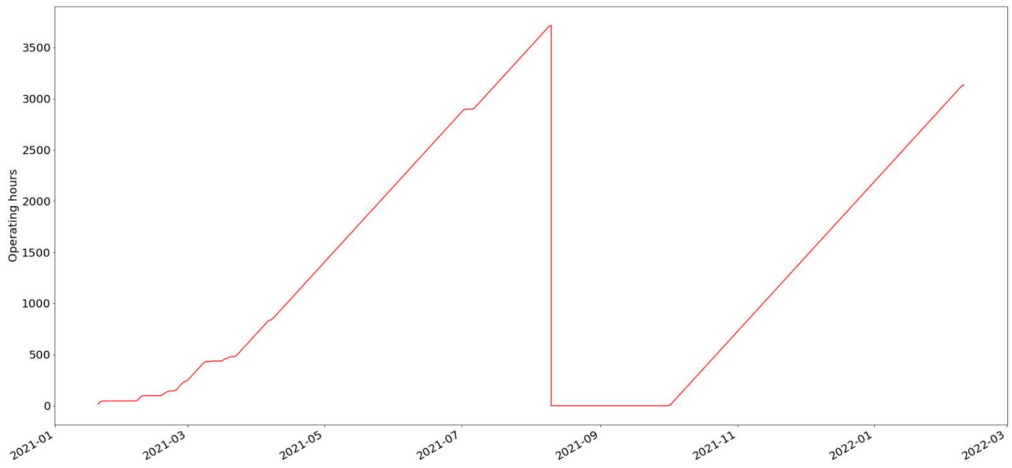
Calculated lamps` operating time of the 1st chromatograph (First approach).

Source: [authors research]

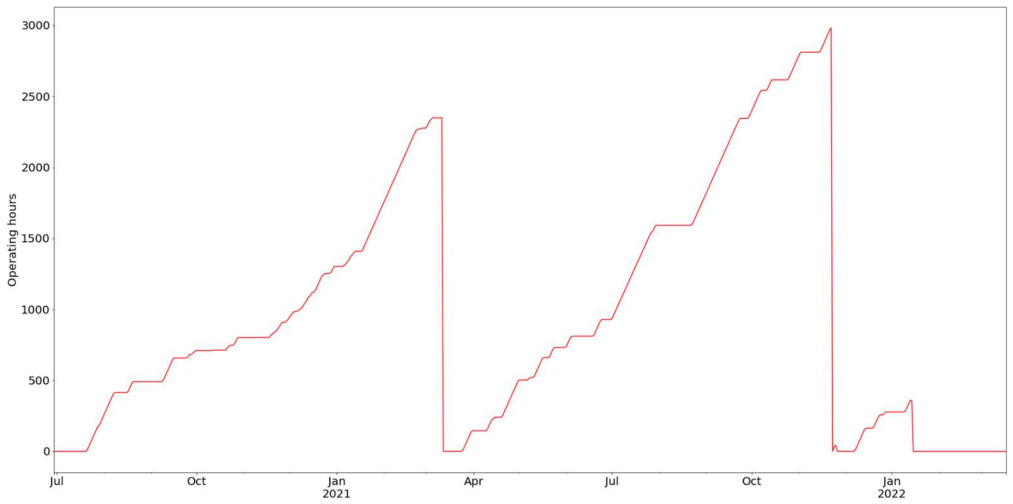


Calculated lamps` operating time of the 2nd chromatograph (First approach).

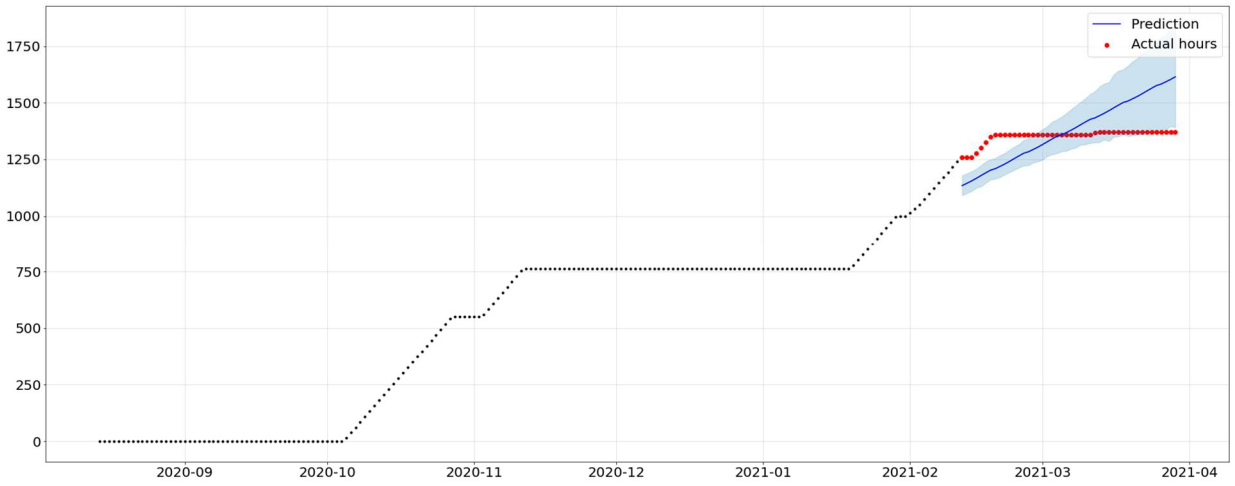
Source: [authors research]



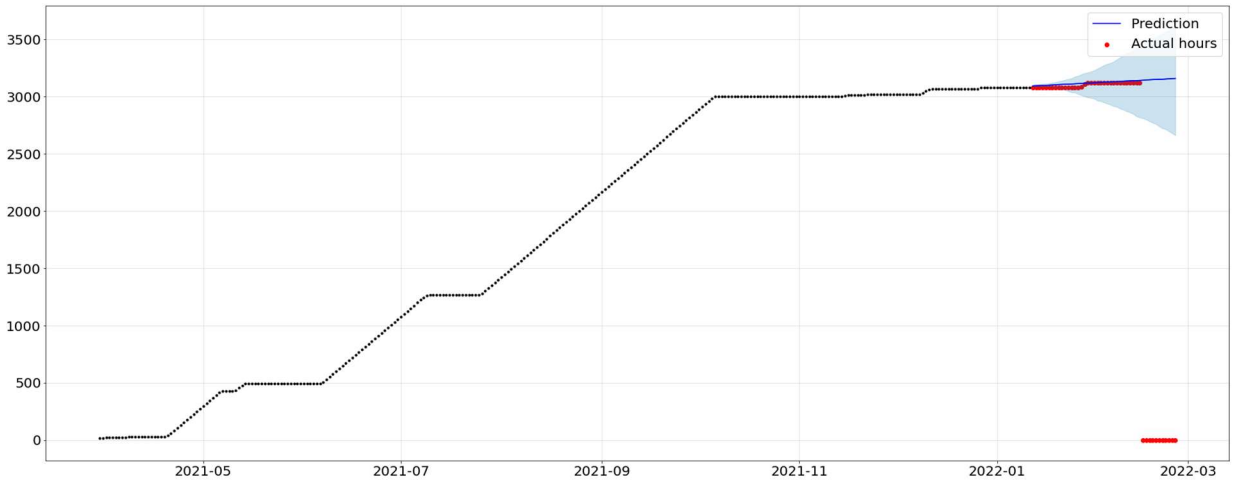
Calculated lamps` operating time of the 3d chromatograph (First approach). Source: [authors research]



Calculated lamps` operating time of the 5th chromatograph (First approach). Source: [authors research]



Regression model for prediction of the lamp`s operating time (Chromatograph 0, lamp 1). Source: [authors research]

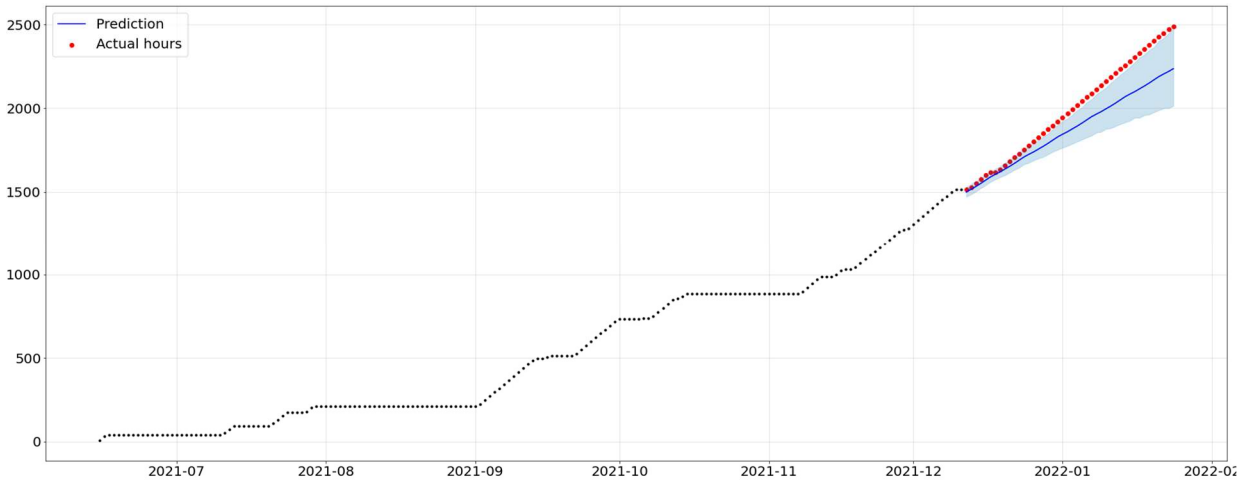


Regression model for prediction of the lamp`s operating time (Chromatograph 0, lamp 2).

Source: [authors research]

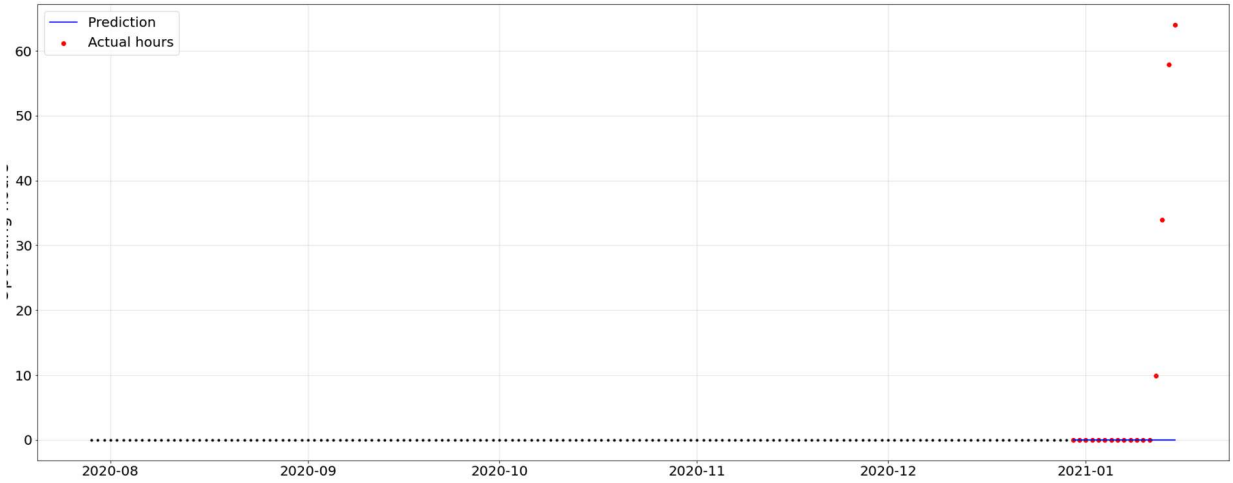
Regression model for prediction of the lamp`s operating time (Chromatograph 1, lamp 2).

Source: [authors research]



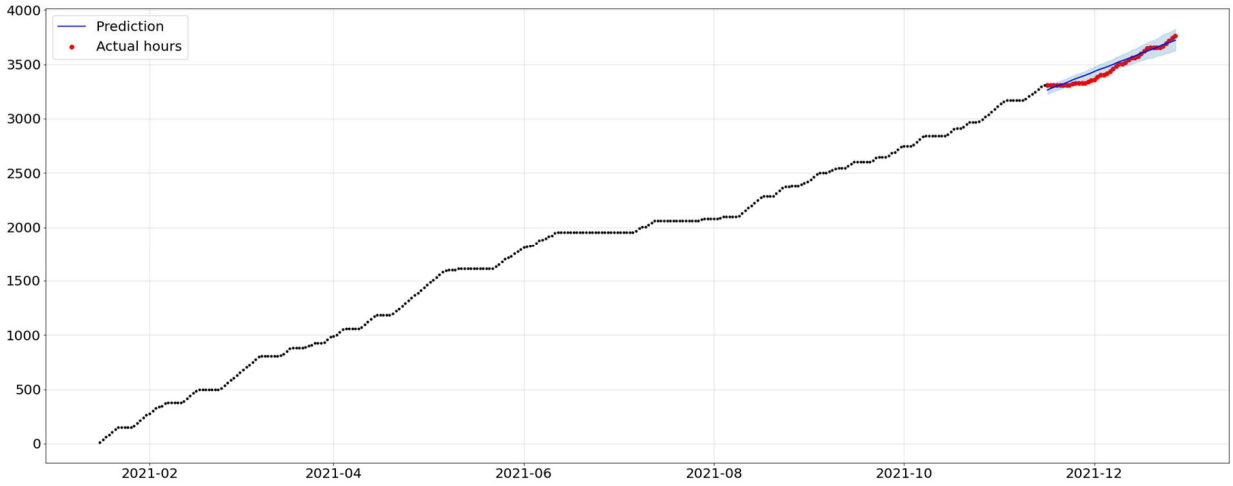
Regression model for prediction of the lamp`s operating time (Chromatograph 1, lamp 3).

Source: [authors research]



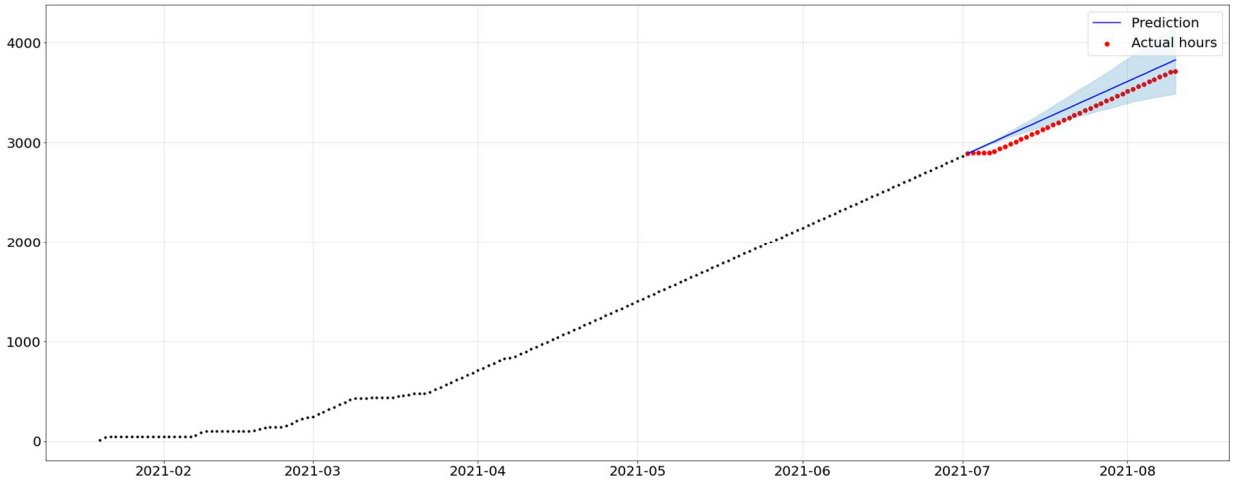
Regression model for prediction of the lamp`s operating time (Chromatograph 2, lamp 1).

Source: [authors research]



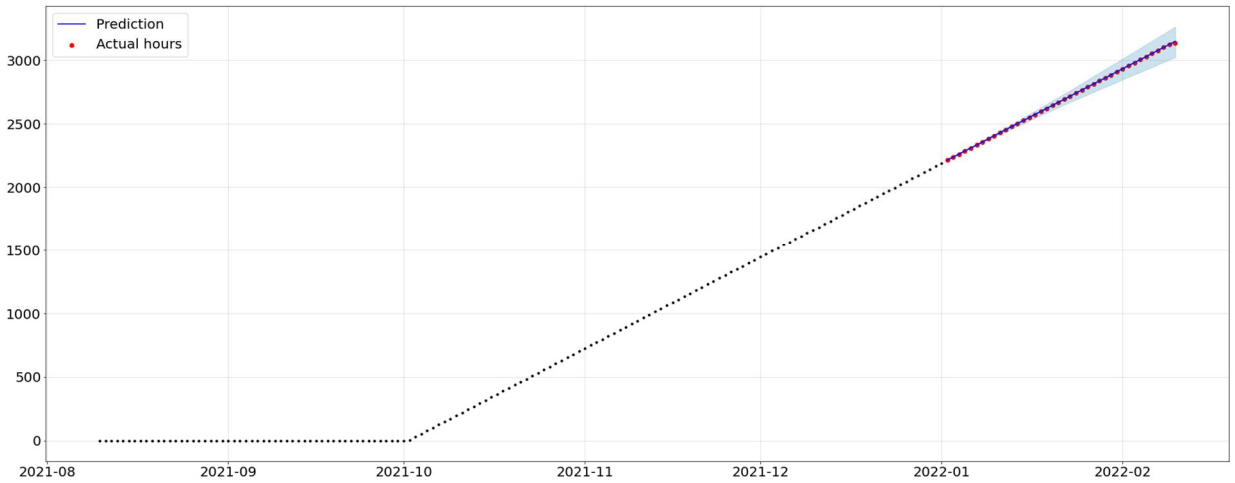
Regression model for prediction of the lamp`s operating time (Chromatograph 2, lamp 2).

Source: [authors research]



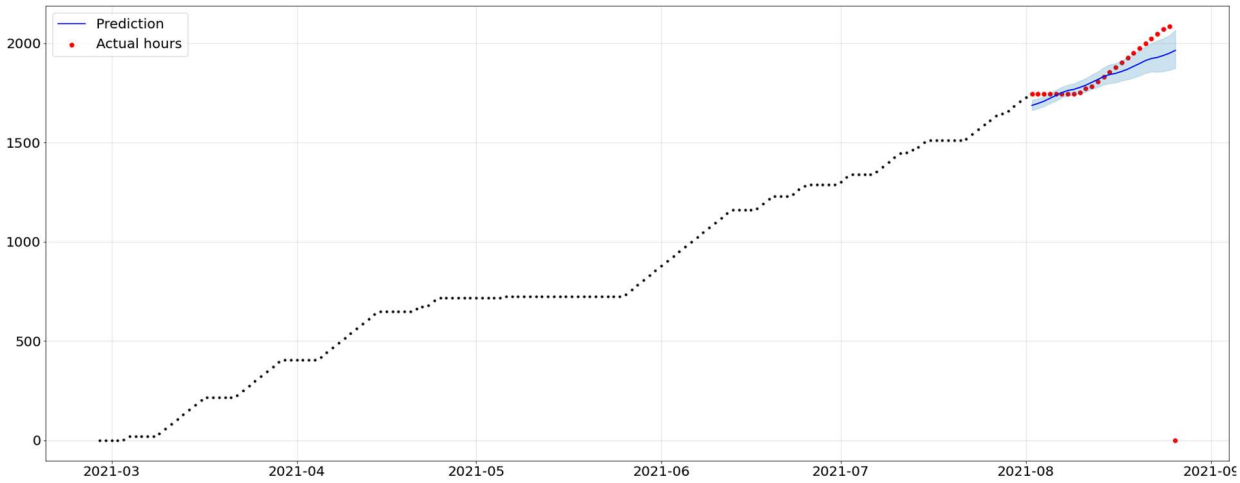
Regression model for prediction of the lamp`s operating time (Chromatograph 3, lamp 1).

Source: [authors research]



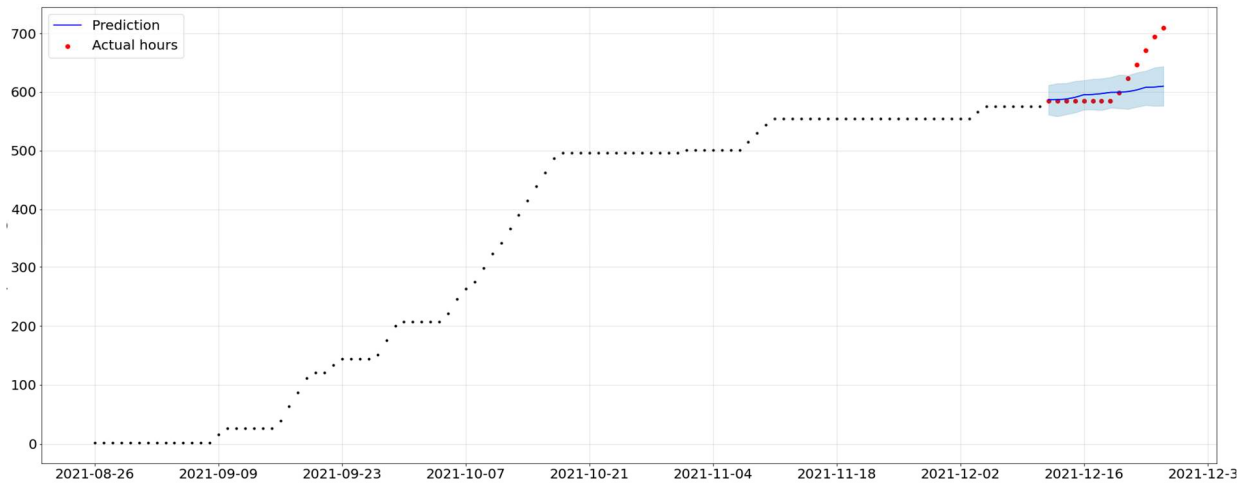
Regression model for prediction of the lamp`s operating time (Chromatograph 3, lamp 2).

Source: [authors research]



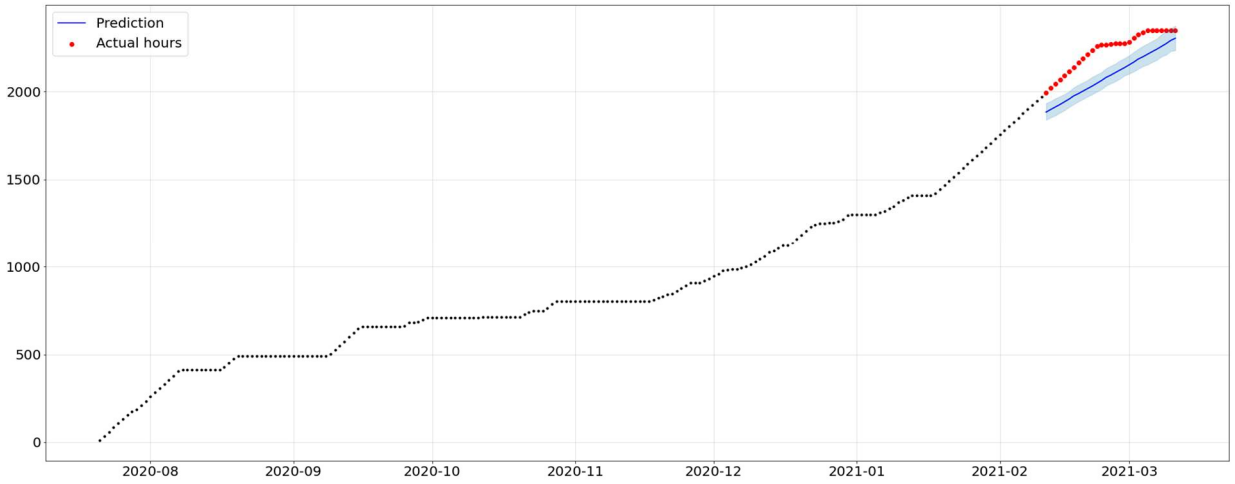
Regression model for prediction of the lamp`s operating time (Chromatograph 4, lamp 2).

Source: [authors research]



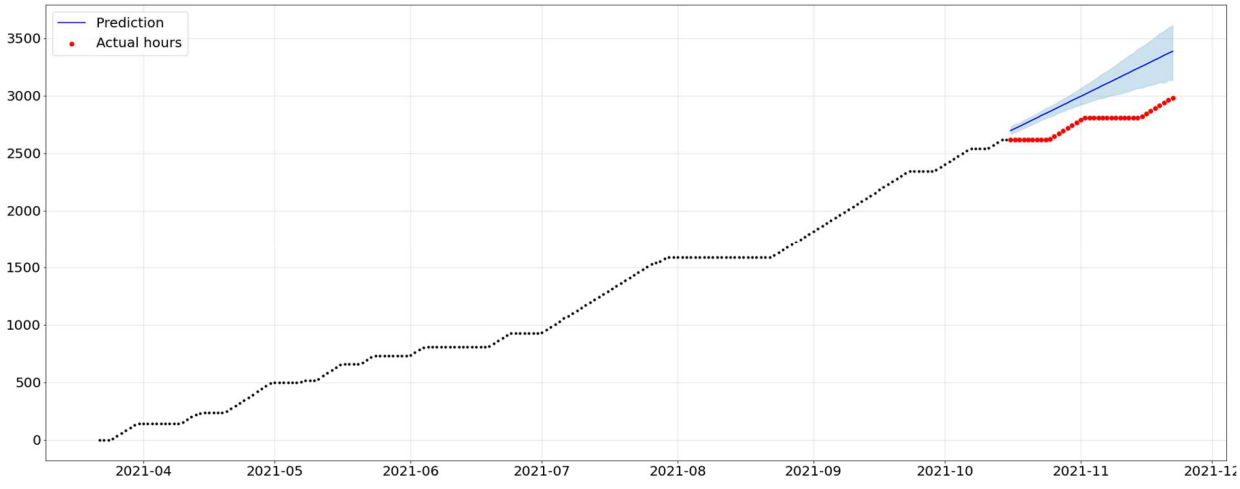
Regression model for prediction of the lamp`s operating time (Chromatograph 4, lamp 3).

Source: [authors research]



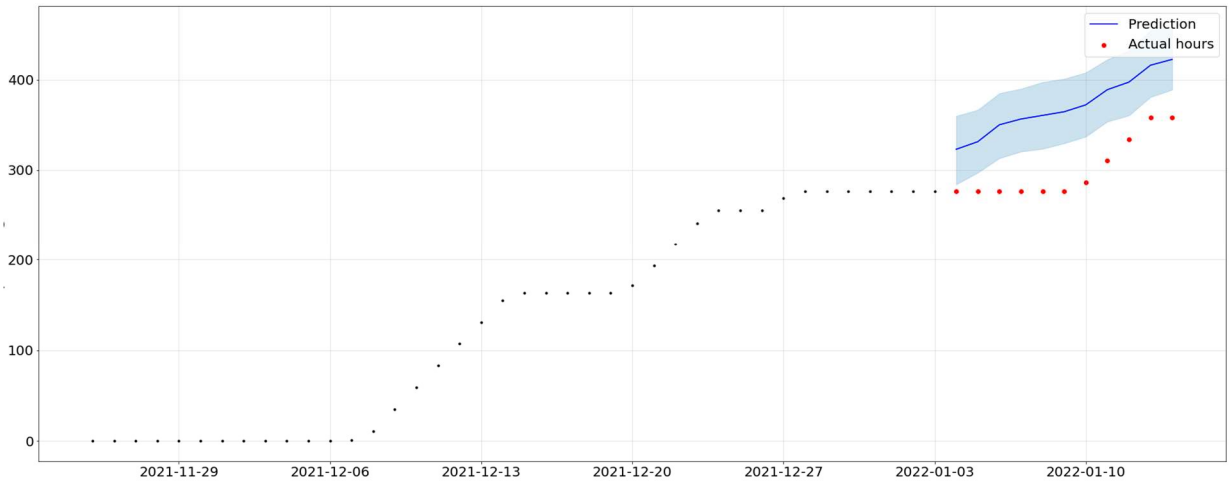
Regression model for prediction of the lamp`s operating time (Chromatograph 5, lamp 1).

Source: [authors research]



Regression model for prediction of the lamp`s operating time (Chromatograph 5, lamp 2).

Source: [authors research]



Regression model for prediction of the lamp`s operating time (Chromatograph 5, lamp 4).

Source: [authors research]

		dataset0		dataset1		dataset2	dataset3		dataset4		dataset5		dataset6	dataset7
		29/03/21 - 17/12/21	25/02/22 - 12/03/22	15/06/21 - 23/12/21	24/12/21 - 29/12/22	15/01/21 - 23/12/21	02/10/21 - 15/12/21	10/02/22 - 17/03/22	26/08/21 - 21/12/21	25/12/21 - 24/02/22	22/03/21 - 21/12/21	24/12/21 - 17/03/22	24/12/21 - 24/02/22	24/12/21 - 17/01/22
Model №	**Actual working hours**	**2582**	**28**	**1626,7**	**1228,3**	**1771,5**	**2266,7**	**702**	**1054,7**	**623,3**	**244,7**	**708,3**	**59,17**	**743,61**
model1	**Calc**	1953,19	177,21	2362,64	120,21	4230,4	459,21	845,8	1149,2	1416,44	3490,81	606,25	1,44	29,46
	error	-24,35	532,89	45,24	-90,21	138,8	-79,74	20,48	8,96	127,25	1326,57	-14,41	-97,57	-96,04
model2	**Calc**	2527,29	177,21	2586,4	120,21	5011,84	1707,07	845,8	1212,41	1416,44	4385,08	1235,34	1,44	29,46
	error	-2,12	532,89	59	-90,21	182,92	-24,69	20,48	14,95	127,25	1692,02	74,41	-97,57	-96,04
model3	**Calc**	1799,77	98,6	2013,22	120,21	3872,87	459,21	845,8	1118,09	1416,44	3543,53	651,35	1,44	29,82
	error	-30,3	252,14	23,76	-90,21	118,62	-79,74	20,48	6,01	127,25	1348,11	-8,04	-97,57	-95,99
model4	**Calc**	2213,76	98,6	3196,58	120,21	4443,16	1707,07	845,8	1212,66	1416,44	4581,27	1579,95	1,44	29,82
	error	-14,26	252,14	96,51	-90,21	150,81	-24,69	20,48	14,98	127,25	1772,2	123,06	-97,57	-95,99
model5	**Calc**	1476,22	100,12	1625,38	55,63	2854,8	52,21	845,8	742,27	23,75	2482,15	581,01	1,44	29,87
	error	-42,83	257,57	-0,08	-95,47	61,15	-97,7	20,48	-29,62	-96,19	914,36	-17,97	-97,57	-95,98
model6	**Calc**	2082,28	23,96	3163,81	122,87	4127,17	1300,08	845,8	983,79	23,75	3956,34	83,15	1,44	29,87
	error	-19,35	-14,43	94,49	-90	132,98	-42,64	20,48	-6,72	-96,19	1516,81	-88,26	-97,57	-95,98

		dataset0		dataset1		dataset2	dataset3		dataset4		dataset5		dataset6	dataset7
		29/03/21 - 17/12/21	25/02/22 - 12/03/22	15/06/21 - 23/12/21	24/12/21 - 29/12/22	15/01/21 - 23/12/21	02/10/21 - 15/12/21	10/02/22 - 17/03/22	26/08/21 - 21/12/21	25/12/21 - 24/02/22	22/03/21 - 21/12/21	24/12/21 - 17/03/22	24/12/21 - 24/02/22	24/12/21 - 17/01/22
model7	**Calc**	1542	100,12	1648,42	55,63	3156,46	52,21	845,8	756,24	23,75	2613	587,04	1,44	29,87
	error	-40,28	257,57	1,34	-95,47	78,18	-97,7	20,48	-28,3	-96,19	967,84	-17,12	-97,57	-95,98
model8	**Calc**	2148,06	23,96	3186,85	122,87	5351,84	1300,08	845,8	997,76	23,75	4672,98	1386,18	1,44	29,87
	error	-16,81	-14,43	95,91	-90	202,11	-42,64	20,48	-5,4	-96,19	1809,68	95,71	-97,57	-95,98
model9	**Calc**	1508,3	100,11	1215,37	55,63	2923,19	52,21	845,8	511,87	23,75	2278,68	586,54	1,44	7,45
	error	-41,58	257,54	-25,29	-95,47	65,01	-97,7	20,48	-51,47	-96,19	831,21	-17,19	-97,57	-99
model10	**Calc**	1850,1	23,95	1865,04	122,87	4705,28	1300,08	845,8	689,56	23,75	3934,66	1385,68	1,44	7,45
	error	-28,35	-14,46	14,65	-90	165,61	-42,64	20,48	-34,62	-96,19	1507,95	95,63	-97,57	-99

Results of calculation of working hours using the approach 2. Source: [authors research]

```
#modules import
import pandas as pd
from collections import defaultdict
import datetime
import re
from datetime import timedelta
import matplotlib.pyplot as plt
import numpy as np
from tqdm import tqdm
pd.options.mode.chained_assignment = None  # default='warn'
import statistics
import os
import seaborn as sns
from prophet import Prophet
from sklearn.metrics import mean_squared_error, mean_absolute_error
import copy
```

```
#set constant
on_top= True              # collapse messages 'lamp on' to the 1st one. False - to the last one
off_top = True            # collapse messages 'lamp off' to the 1st one. False - to the last one
enable_ready = True       # take into account messages "get system ready"
enable_shutdown = True    # take into account messages "shutdown"
enable_s_start = True     # take into account messages "s started"
enable_s_stop = True      # take into account messages "s stopped" or "s aborted"
max_hours_to_restart = 0  # number of working hours calculated after which the calculation starts from 0, for model 11.
time_sleep = 24*3600      # time with no logs which will be considered as equipment shutdown
critical_lost_msg = 0.25  # max allowed deviation of relations on/off messages from 1, higher deviation generates alert messages
prophet_is_test = False   # prediction mode. when True - predict for past periods of time (needs to be defined)
#to find error metrics. When False - predict for future perios of time based on the historic activity from the last period
min_days_to_prophet = 30   # min number of das that should be in the last period to predict the future
max_hours_to_change_lamp = 3000  # working hours limit for each model produces the recommendation to change the lamp
upd = False                # cumulative working hours will be updated to real values at least for 1 dataset
actual_hours = 2855       # set only if it's necessary to update amount of working hours for the last subset (the current one)
#otherwise set this value to 0. If it's necessary to update values of past periods, enter them in chapter visualisation.

dataset = 1
filename = str(dataset)+'.csv' #name of the file with data
result_path = './results/'+str(dataset)+'/'
seq_num = 11 #set of constants
graph_on = True #False #True #if we plot graphs of number of different messages and recalculate related health parameters
if not os.path.exists(result_path):
        os.makedirs(result_path)
```

Python code. Import modules and set constants. Source: [authors research]

```
#dataframes import
df0 = pd.DataFrame(pd.read_csv(filename))
```

```
df0.head(5)
```

	date	event_class	text_message
0	2015-03-02 13:08:18.000	log.from.chromatograph	1 G1321A:DE60557600 - UV Lamp switched on.
1	2015-03-02 13:08:18.000	log.from.chromatograph	1 G1321A:DE60557600 - Lamp on
2	2015-03-02 13:14:15.000	log.from.chromatograph	S started
3	2016-11-16 14:13:45.000	log.from.chromatograph	1 G1316A:DEACN47005 - Thermostat off
4	2016-11-16 16:33:09.000	log.from.chromatograph	1 G1311B:DEAE415582 - Turn pump on.

```
len(df0)
```

```
9808
```

```
df0.describe()
```

	date	event_class	text_message
count	9808	9808	9808
unique	6820	9	248
top	2021-11-19 13:07:41.000	log.from.chromatograph	S started
freq	13	7802	648

Python code. Exploratory data analysis. Source: [authors research]

```
#dataframe preparation
df0 = df0.rename({'text_message;;;;;;;;;': 'text_message'}, axis='columns')
df0['text_message'] = df0['text_message'].str.lower() #transforming text messages to lower case
df0['text_message'] = df0['text_message'].astype(str) #transforming text messages to string type
for i in range(0, len(df0)):
    df0["date"][i]= datetime.datetime.strptime(df0.loc[i]['date'][:23] , '%Y-%m-%d %H:%M:%S.%f')
    #df0["date"][i]= datetime.datetime.strptime(df0.loc[i]['date'] , '%Y-%m-%d %H:%M:%S')
    df0["text_message"][i] = df0["text_message"][i].replace (';', '')
df0['time_diff'] = timedelta(0)
for i in range(1, len(df0)):
    df0.loc[i,'time_diff'] =  df0.loc[i]['date'] - df0.loc[i-1]['date']
df0["flag"]=0
```

```
#creating list of indexes with timestamps for beginning and end of calculations
index_list = [len(df0)] #creating initial list of indexes
date_24dec= datetime.datetime.strptime('2021-12-24', '%Y-%m-%d')
#i_23dec = df0.loc[df0.date>=date_24dec].head(1).index.tolist() #calculating index of the 1st log with date more than 24 Dec
#index_list = index_list +  i_23dec #adding intermediate indexes to the text
for i in range(0, len(df0)-1):
    if bool(re.findall(r'.*замена\s*уф-ламп.*', df0.loc[i]['text_message'])) and\
    not bool(re.findall(r'.*замена\s*уф-ламп.*', df0.loc[i+1]['text_message'])):  #which are not followed by the same service me
        print (i, df0.loc[i]['date'],df0.loc[i]['text_message'])
        index_list.append(i) #adding indices of service messages to the list of indices
index_list.sort()
index_list
```

```
8111 2021-06-15 14:20:37 отказ: по непредвиденной причине выполненные работы: замена уф-лампы

[8111, 9808]
```

Python code. Data Preprocessing. Source: [authors research]

```
#calculation of max and average time of "equipment sleep"
for i in range(0,len(index_list)-1):
    df_health.loc[i]['max_sleep']= max(df_sets[i]['time_diff'])
```

Python code. Calculation of max time without messages. Source: [authors research]

```
for i in range(0,len(index_list)-1):
    ind_lampon = 0
    ind_lampoff = 0
    ind_error = 0
    ind_ready = 0
    ind_shutdown = 0
    ind_start = 0
    ind_abort = 0
    ind_complete= 0


    for j in range(0, len(df_sets[i])):
        if bool(re.findall(r'.*lamp\s*on.*', df_sets[i].loc[j]['text_message'])) and\
        bool(re.findall(r'[^vis]', df_sets[i].loc[j]['text_message'])):
            ind_lampon +=1

        elif bool(re.findall(r'.*lamp\s*off.*', df_sets[i].loc[j]['text_message'])) and\
        bool(re.findall(r'[^vis]', df_sets[i].loc[j]['text_message'])):
            ind_lampoff +=1

        elif bool(re.findall(r'error', df_sets[i].loc[j]['text_message'])):
            ind_error +=1

        elif bool(re.findall(r'.*lamp\s*off.*', df_sets[i].loc[j]['text_message'])):
            ind_error +=1

        elif bool(re.findall(r'.*get\s*system\s*ready.*', df_sets[i].loc[j]['text_message'])):
            ind_ready +=1

        elif bool(re.findall(r'shutdown', df_sets[i].loc[j]['text_message'])):
            ind_shutdown +=1

        elif bool(re.findall(r'.*s\s*start.*', df_sets[i].loc[j]['text_message'])):
            ind_start +=1

        elif bool(re.findall(r'.*s\s*abort.*', df_sets[i].loc[j]['text_message'])):
            ind_abort +=1

        elif bool(re.findall(r'.*s\s*complete.*', df_sets[i].loc[j]['text_message'])):
            ind_complete +=1

    ind = [ind_lampon, ind_lampoff, ind_error,ind_ready,ind_shutdown, ind_start, ind_abort, ind_complete]

    df_health.loc[i]['msg_on/msg_off'] = 0
    df_health.loc[i]['msg_ready/msg_shutdown'] = 0
    df_health.loc[i]['msg_start/msg_stop'] = 0
    df_health.loc[i]['msg_error'] = ind_error
```

Python code. Calculation of different types of messages. Source: [authors research]

```
if graph_on == True:#check if we need to show graphs, default is True
    x = range(len(ind))
    plt.figure(figsize=(15,10))
    ax = plt.gca()
    ax.bar(x, ind, align='center') # align='center' - выравнивание по границе, а не по центру
    plt.title('Dataset ' + str(dataset)+ ' Subset ' + str(i)+ ' '+ \
              str(df_sets[i].loc[0]['date'].date().strftime('%d/%m/%Y')) +\
              '-' + str(df_sets[i].loc[len(df_sets[i])-1]['date'].date().strftime('%d/%m/%Y')))
    plt.ylabel('Number of messages')
    plt.xlabel('Text message')
    ax.set_xticks(x)
    ax.set_xticklabels(('lamp on', 'lamp off', 'error','system ready', 'shutdown', 's start', 's abort', 's complete'))
    autolabel(ax.patches, ind, height_factor=1.01)
    plt.plot()
    #plt.savefig(result_path +'subset_'+ str(i)+'.plot_msg.png')
```

Python code. Graph of number of different messages generation. Source: [authors research]

```
#addition flags for the work start
for i in range(0,len(index_list)-1):
    length = range(0, len(df_sets[i]))
    for j in tqdm(length):
        if bool(re.findall(r'.*lamp\s*on.*', df_sets[i].loc[j]['text_message'])) and\
        bool(re.findall(r'[^vis]', df_sets[i].loc[j]['text_message'])):
            df_sets[i]['flag'][j] = 19  #flag 19 for all "lamp on" messages

        if bool(re.findall(r'.*get\s*system\s*ready.*', df_sets[i].loc[j]['text_message'])) and\
        enable_ready: #default value is True
            df_sets[i]['flag'][j] = 18 #flag 18 for all "get system ready" messages

        if bool(re.findall(r'.*s\s*start.*', df_sets[i].loc[j]['text_message'])) and\
        enable_s_start: #default value is True
            df_sets[i]['flag'][j] = 17 #flag 17 for all "S started" messages

        if  j!=0 and df_sets[i]['flag'][j] == 0 and bool(re.search(r'[а-яА-ЯёЁ]', df_sets[i].loc[j]['text_message'])) == False:
            if bool(re.search(r'[а-яА-ЯёЁ]', df_sets[i].loc[j-1]['text_message'])):
                df_sets[i]['flag'][j] = 16 #flag 16 for all messages following service text messages which are not service messa

            elif df_sets[i].loc[j]['time_diff'].total_seconds()> time_sleep: #default value is 24 hours
                df_sets[i]['flag'][j] = 15 #flag 15 for all messages which have more than preset time lap with the previous one

#addition flags for the work finish
        if bool(re.findall(r'.*lamp\s*off.*', df_sets[i].loc[j]['text_message'])) and\
        bool(re.findall(r'[^vis]', df_sets[i].loc[j]['text_message'])):
            df_sets[i]['flag'][j] = 29  #flag 29 for all "lamp off" messages

        if bool(re.findall(r'shutdown', df_sets[i].loc[j]['text_message']))and\
        enable_shutdown: #default value is True:
            df_sets[i]['flag'][j] = 28 #flag 28 for all "shutdown" messages

        if j !=len(df_sets[i])-1 and bool(re.search(r'[а-яА-ЯёЁ]', df_sets[i].loc[j]['text_message'])) == False:
            if bool(re.search(r'[а-яА-ЯёЁ]', df_sets[i].loc[j + 1]['text_message'])):
                df_sets[i]['flag'][j] = 27 #flag 27 for all messages before service text messages which are not service messages

            elif df_sets[i].loc[j+1]['time_diff'].total_seconds()> time_sleep: #default value is 24 hours
                df_sets[i]['flag'][j] = 26 #flag 26 for all messages which have more than preset time lap with the next one

        if df_sets[i]['flag'][j] != 0 and\
        (bool(re.findall(r'.*s\s*abort.*', df_sets[i].loc[j]['text_message'])) or\
        bool(re.findall(r'.*s\s*complete.*', df_sets[i].loc[j]['text_message']))) and\
        enable_s_stop: #default value is True
            df_sets[i]['flag'][j] = 25 #flag 25 for all "s abort or s completed" messages
#and df_sets[i]['flag'][j] == 0
```

Python code. Addition of start and stop flags. Source: [authors research]

```
#addition of the 1st and the last messages to consider cases when the 1st flaf is "off" or the last flag is "on"
for i in range(0,len(index_list)-1):
    length = len(df_sets[i])
    first_row = {'date':df_sets[i].loc[0]['date'], 'event_class':'', 'text_message':'', 'time_diff':timedelta(0), 'flag':10}
    last_row = {'date':df_sets[i].loc[length-1]['date'], 'event_class':'', 'text_message':'', 'time_diff':timedelta(0),'flag':20
    df_sets[i].loc[-1] = first_row  # adding the 1st row
    df_sets[i].index = df_sets[i].index + 1
    df_sets[i] = df_sets[i].sort_index()  #
    df_sets[i] =df_sets[i].append(last_row, ignore_index=True)   # adding the last row
```

```
#creating new dataframes with only non zero flags and transforming them by collapsing flags of the same type
df_flags = {}
df_clean = {}
for i in range(0,len(index_list)-1):
    df_flags[i] = pd.DataFrame()
    df_flags[i] = df_sets[i].loc[df_sets[i]["flag"] !=0].reset_index(drop=True)
    ind_flags = df_flags[i].index.tolist()
    lind = len(ind_flags)
    j = 0

    while j < lind - 1:
        if df_flags[i].loc[ind_flags[j]].flag // 10 == df_flags[i].loc[ind_flags[j+1]].flag // 10:
            if df_flags[i].loc[ind_flags[j]].flag == df_flags[i].loc[ind_flags[j+1]].flag:
                if df_flags[i].loc[ind_flags[j]].flag // 10 == 1:
                    ind_to_drop = j+on_top #if on_top = True, the 2nd "on" flag will be eliminated
                else:
                    ind_to_drop = j+off_top #if off_top = True, the 2nd "off" flag will be eliminated
            elif df_flags[i].loc[ind_flags[j]].flag > df_flags[i].loc[ind_flags[j+1]].flag:
                ind_to_drop = j+1
            else:
                ind_to_drop = j
            ind_flags.remove(ind_flags[ind_to_drop])
            lind -= 1
        else:
            j +=1
    df_clean[i] = df_flags[i][df_flags[i].index.isin(ind_flags)].reset_index(drop=True)
    df_clean[i]['time_diff'] = timedelta(0)
    for j in range(1, len(df_clean[i]),2):
        df_clean[i].loc[j,'time_diff'] = df_clean[i].loc[j,'date'] - df_clean[i].loc[j-1,'date']
```

Python code. Database transformation depending on flags values. Source: [authors research]

```
#adding a new column with only dates to all subsets
for i in range (0,len(index_list)-1):
    df_clean[i]['only_date'] = ''
    for j in range(0,len(df_clean[i])):
        df_clean[i]['only_date'][j] = df_clean[i].loc[j]['date'].date().strftime('%d/%m/%Y')

#forming a dictionary of dataframes with number of working hours per day
lind = list(range(0,len(index_list)-1))
d_dates = dict.fromkeys(lind)

for i in range(0,len(index_list)-1):
    dates = pd.date_range(df_clean[i].loc[0]['date'].date(), df_clean[i].loc[len(df_clean[i])-1]['date'].date()).strftime('%d/%m
    #print(dates)
    df_dates = pd.DataFrame(index = dates, columns = ['dates','wh', 'calc_wh','wh_sum'])
    #print(df_dates)
    df_dates['wh'] = timedelta(0)
    #df_dates['wh_sum'] = timedelta(0)
    #df_dates['calc_wh'] = float(0)
    #df_dates['wh_sum'] = 0
    lind2 = len(df_clean[i])

    for j in range(0, lind2-1,2):
        idx = df_clean[i].loc[j]['only_date']
        if df_clean[i].loc[j]['only_date'] == df_clean[i].loc[j+1]['only_date']:
            df_dates.at[idx,'wh'] +=df_clean[i].loc[j+1]['time_diff']
#           print(df_clean[i].loc[j]['only_date'])
        else:
            base = df_clean[i].loc[j]['date'].date() + datetime.timedelta(days=1)
            df_dates.at[idx,'wh'] += datetime.datetime.combine(base, datetime.time.min) - df_clean[i].loc[j]['date']
            #base += datetime.timedelta(days=1)
            while base < df_clean[i].loc[j+1]['date'].date():
                df_dates.at[base.strftime('%d/%m/%Y'),'wh'] += timedelta(hours=24)
                base += datetime.timedelta(days=1)
            df_dates.at[base.strftime('%d/%m/%Y'),'wh'] += df_clean[i].loc[j+1]['date'] - datetime.datetime.combine(base, dateti

    for date in dates:
        df_dates.at[date,'calc_wh'] = round(df_dates.loc[date]['wh'].total_seconds()/3600,2)
    df_dates['wh_sum'] = df_dates['calc_wh'].cumsum()
    df_dates['dates'] = pd.to_datetime(df_dates.index,format = "%d/%m/%Y")
    df_dates.reset_index(drop=True, inplace=True)
    d_dates[i] = df_dates
     #print(d_dates[i] )
```

```
#creating graphs of cumulative value of working hours for each subset
for i in range(0,len(index_list)-1):
    plt.plot(d_dates[i]['dates'], d_dates[i]['wh_sum'], label="working hours")
    #plt.ylim(0,4000)
    plt.xlim(min(d_dates[i]['dates']),max(d_dates[i]['dates']))
    plt.ylabel ('Hours_total', fontsize=16)
    plt.xlabel ('Days', fontsize=16)
    plt.title('Dataset: ' + str(dataset)+ ' Subset: ' + str(i)+' const_set: '+str(seq_num))
    plt.xticks(fontsize=10)
    plt.yticks(fontsize=10)
    plt.gcf().set_size_inches(10, 5)
    plt.plot()
   # plt.gcf().savefig(result_path+'subset_'+ str(i)+'const_set_'+str(seq_num)+ '.plot_hours.png')
    plt.show(block=False)
```

Python code. Transforming database for further visualisation and graphs plotting.

Source: [authors research]

```
#update number of working hours considering real values at time points
if upd:
    d_dates_upd = copy.deepcopy(d_dates) #create copy of dict with dates and working hours per day
    df_offset = pd.DataFrame(index = range(0,len(index_list)-1), columns =['actual_hours','offset'])
    df_offset['actual_hours'] = 0
    df_offset['offset'] = 0
    list_offset = [] #list of subset indices which will be updated
    if actual_hours ==0: #if we set real values for past periods
        df_offset['actual_hours'] = [0,2582,0] #real working hours at the end of the each subset if needed
    else:  #if we apply offset only to the current subset
        df_offset.at[len(df_offset)-1,'actual_hours'] = actual_hours
    for i in range(0,len(index_list)-1):
        if df_offset.loc[i]['actual_hours'] !=0:
            df_offset.at[i,'offset'] = df_offset.loc[i]['actual_hours'] - d_dates[i].iloc[-1]['wh_sum']
            list_offset.append(i) #create list of updated subset numbers
            d_dates_upd[i].at[0,'calc_wh'] += df_offset.at[i,'offset'] #add offset to the 1st date in the subset
            d_dates_upd[i]['wh_sum'] = d_dates_upd[i]['calc_wh'].cumsum()  #recalculate cumulative working hours with offset
```

```
df_offset
```

	actual_hours	offset
0	2855	229

```
#updating graphs taking into account offset to correlate with real values
for i in list_offset:
    plt.plot(d_dates_upd[i]['dates'], d_dates_upd[i]['wh_sum'], label="working hours")
    #plt.ylim(0,4000)
    plt.xlim(min(d_dates_upd[i]['dates']),max(d_dates_upd[i]['dates']))
    plt.ylabel ('Hours_total', fontsize=16)
    plt.xlabel ('Days', fontsize=16)
    plt.title('Dataset: ' + str(dataset)+ ' Subset: ' + str(i)+ ' updated'+' const_set: '+str(seq_num))
    plt.xticks(fontsize=10)
    plt.yticks(fontsize=10)
    plt.gcf().set_size_inches(10, 5)
    plt.plot()
    #plt.gcf().savefig(result_path+'subset_'+ str(i)+'updated'+'const_set_'+str(seq_num)+'.plot_hours.png')
    plt.show(block=False)
```

Python code. Updating graphs considering the real values of working hours.

Source: [authors research]

```
#choose the dict that will be used for further calculations
upd = False
if upd:
    d_calc = copy.deepcopy(d_dates_upd) #updated dict with offsets
else:
    d_calc = copy.deepcopy(d_dates) #initial dict without offsets
```

```
#creating dataframe for working hours and info about constants which were used in calculation
df_hours = pd.DataFrame(index = range(0,len(index_list)-1), columns =['date_start','date_finish','work_hours', 'const_set',\
                                                                      'on_top', 'off_top','enable_ready', 'enable_shutdown',\
                                                                       'enable_s_start', 'enable_s_stop', 'time_sleep'])
for i in range(0,len(index_list)-1):
    df_hours.loc[i]['work_hours'] = s=d_calc[i].iloc[-1]['wh_sum']
    df_hours.loc[i]['date_start'] = d_calc[i].iloc[0]['dates'].date()
    df_hours.loc[i]['date_finish'] = d_calc[i].iloc[-1]['dates'].date()
    #wh_calc = round(df_clean[i]['time_diff'].sum().total_seconds()/3600,2)
    #df_hours.loc[i]['work_hours'] = wh_calc
    #df_hours.loc[i]['date_start'] = df_clean[i].loc[0]['date'].date()
    #df_hours.loc[i]['date_finish'] = df_clean[i].loc[len(df_clean[i])-1]['date'].date()
    df_hours.loc[i]['on_top'] = on_top
    df_hours.loc[i]['off_top'] = off_top
    df_hours.loc[i]['enable_ready'] = enable_ready
    df_hours.loc[i]['enable_shutdown'] = enable_shutdown
    df_hours.loc[i]['enable_s_start'] = enable_s_start
    df_hours.loc[i]['enable_s_stop'] = enable_s_stop
    df_hours.loc[i]['time_sleep'] = time_sleep
    df_hours.loc[i]['const_set'] = seq_num
```

Python code. Creating new dataframe with working hours and constants. Source: [authors research]

```
#calculating average non-stop working days and idle periods
for i in range(0,len(index_list)-1):
    list_work = []
    list_idle = []
    for j in range(0,len(df_clean[i])-1,2):
        time_work = (df_clean[i].loc[j+1]['date'] - df_clean[i].loc[j]['date']).total_seconds()/3600
        list_work.append(time_work)

    for j in range(1,len(df_clean[i])-2,2):
        time_idle = (df_clean[i].loc[j+1]['date'] - df_clean[i].loc[j]['date']).total_seconds()/3600
        list_idle.append(time_idle)
    min_work = min(list_work)
    max_work = max(list_work)
    mean_work = round(np.mean(list_work),2)
    st_dev_work = 0
    if len(list_work) > 1 :
        st_dev_work = round(statistics.pstdev(list_work,mu=mean_work),2)
    mean_idle = round(np.mean(list_idle),2)
    st_dev_idle = 0
    if len(list_idle) > 1 :
        st_dev_idle = round(statistics.pstdev(list_idle),2)
    df_health.loc[i]['avg_idle_hours'] = mean_idle
    df_health.loc[i]['std_idle_hours'] = st_dev_idle
    df_health.loc[i]['avg_nonstop_work_hours'] = mean_work
    df_health.loc[i]['std_nonstop_work_hours'] = st_dev_work
#   print(df_health.loc[i])
#  print(min_work, max_work)
```

```
#form dataframe with working hours, constants and additional indicators for all subsets
if upd:
    df_hours = df_hours.join(df_offset['offset'])
df_final=df_hours.join(df_health)
```

Python code. Calculation of additional indicators and final dataframe formation.

Source: [authors research]

```
#recalculate values if the number of working hours has exceeded 3000 hours (or other pre-defined limit)
if max_hours_to_restart > 0 :
    for i in range(0, len(df_final)):
        if df_final.iloc[i].work_hours > max_hours_to_restart :
            if i !=len(df_final)-1:
                delta = df_final.iloc[i].work_hours - max_hours_to_restart
                df_final.at[i+1,'work_hours'] +=delta
            df_final.at[i, "work_hours"] -= max_hours_to_restart
            print("Warning! Work hours has exceeded the set! (start date: {0}, end date: {1})".format(df_final.iloc[i].date_star
```

Python code. Recalculation of working hours if the total amount has exceeded the pre-defined value. Source: [authors research]

```
#generate warning messages
working_on = df_clean[len(df_clean) - 1].iloc[-1].flag < 20
print("Lamp is working: {0}".format(working_on))

check_date = df_clean[len(df_clean) - 1].iloc[-1].date
period_to_now = datetime.datetime.now() - check_date
hours_to_now = round(period_to_now.days * 24 + period_to_now.seconds / 60 / 60,2)

if working_on :
    if hours_to_now > df_final.iloc[-1].avg_nonstop_work_hours :
        print("Warning! Work hours has exceeded the average non-stop hours! (value: {0}, avg: {1})".format(hours_to_now, df_fina
else :
    if hours_to_now > df_final.iloc[-1].avg_idle_hours :
        print("Warning! Idle hours has exceeded the average idle hours! (value: {0}, avg: {1})".format(hours_to_now, df_final.il


val_on_off = abs(1 - df_health.iloc[-1]['msg_on/msg_off'])
if val_on_off > critical_lost_msg :
    print("Warning! Messages on/off has exceeded the set! (subset: {2}, value: {0}, set: {1})".format(val_on_off, critical_lost_

value_ready_shutdown = abs(1 - df_health.iloc[-1]['msg_ready/msg_shutdown'])
if value_ready_shutdown > critical_lost_msg :
    print("Warning! Messages ready/shutdown has exceeded the set! (subset: {2}, value: {0}, set: {1})".format(value_ready_shutdo

value_start_stop = abs(1 - df_health.iloc[-1]['msg_start/msg_stop'])
if value_start_stop > critical_lost_msg :
    print("Warning! Messages start/stop has exceeded the set! (subset: {2}, value: {0}, set: {1})".format(value_start_stop, crit
```

Python code. Warning messages generation in case of possible problems with data. Source: [authors research]

```python
#MAPE function
def mean_absolute_percentage_error(y_true, y_pred):
    y_true, y_pred = np.array(y_true), np.array(y_pred)
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100
```

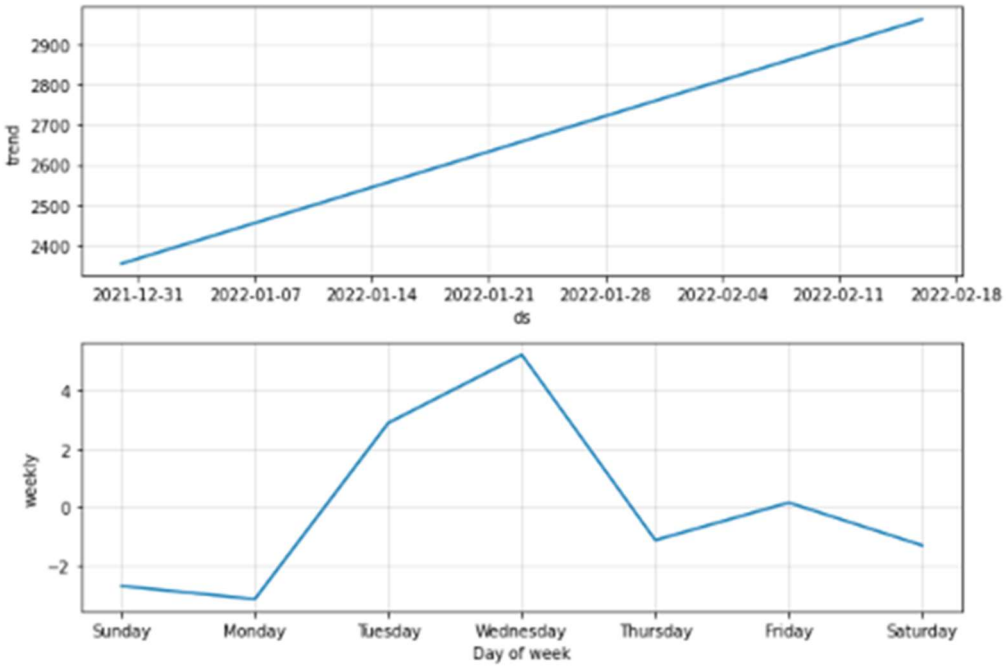
```python
# make preductions

if prophet_is_test:
    subset_id = 1 #set manually the id for which prediction will be conducted
else:
    subset_id = len(index_list)-2 #to predict the future the subset is automatically set to be the last one
#test!!!!!
#subset_id = 1
#prophet_is_test = False
#fewDataToProphet = False
#max_hours_to_change_lamp = 2400

split_date = d_calc[subset_id].iloc[-1].dates
date_to_change = d_calc[subset_id].iloc[-1].dates
numDays = d_dates[subset_id].iloc[-1].dates - d_dates[subset_id].iloc[0].dates
fewDataToPredict = len(d_calc[subset_id]) < min_days_to_prophet #check if historic activity is enough to predict future
train = {}
test = {}
test_fcst = {}
model = Prophet(changepoint_prior_scale=0.001, daily_seasonality=False, yearly_seasonality=False)

df_final.at[subset_id, 'mse']  = 0
df_final.at[subset_id, 'mae']  = 0
df_final.at[subset_id, 'mape'] = 0
df_final.at[subset_id, 'prodicted_date']    = 0
df_final.at[subset_id, 'predicted_value']   = 0
df_final.at[subset_id, 'date_to_change']  = 0
df_final.at[subset_id, 'value_to_change'] = max_hours_to_change_lamp
```

Python code. Preparation for working hours prediction. Source: [authors research]

```
if fewDataToPredict :
    print("Warning! Dataset contains too few data to predict!")

else :
    if prophet_is_test:
        # 80% of dates range is the train data and 20% is the test data
        split_date = d_calc[subset_id].iloc[0].dates + (numDays * 0.8)
        train = d_calc[subset_id].loc[d_calc[subset_id].dates <= split_date].copy()
        test  = d_calc[subset_id].loc[d_calc[subset_id].dates > split_date].copy()
        print("Prophet in TEST mode!")
    else:
        # whole date range is the train data (as 80%) and we will create empty array of the test data (20%)
        train = d_calc[subset_id].copy()
        test  = train[0:0] # getting only the structure of train dataframe
        num = int(len(d_calc[subset_id]) *  0.25)
        test_end = d_calc[subset_id].iloc[-1].dates

        for i in range(1, num + 1) :
            test = test.append({'dates' : test_end + timedelta(days=i), 'wh': 0, 'calc_wh': 0, 'wh_sum' : d_calc[subset_id].iloc

    # Preparing train and test dataframes to prophet
    train = train.rename(columns={'dates':'ds', 'wh_sum':'y'})
    test  = test.rename(columns={'dates':'ds', 'wh_sum':'y'})

    model.fit(train)
    test_fcst = model.predict(test)

    # calculating errors
    mse  = mean_squared_error (y_true = test['y'], y_pred = test_fcst['yhat'])
    mae  = mean_absolute_error(y_true = test['y'], y_pred = test_fcst['yhat'])
    mape = mean_absolute_percentage_error(test['y'], test_fcst['yhat'])

    # add info to the final table
    if prophet_is_test :
        df_final.at[subset_id, 'mse']  = round(mse, 3)
        df_final.at[subset_id, 'mae']  = round(mae, 3)
        df_final.at[subset_id, 'mape'] = round(mape, 3)

    df_final.at[subset_id, 'predicted_date']  = test.iloc[-1].ds
    df_final.at[subset_id, 'predicted_value'] = round(test_fcst.iloc[-1].yhat_upper, 0)

    # if predicted value is greater than the limit of working hours for the lamp then calculate the date to change UV lamp
    if test_fcst.iloc[-1].yhat_upper > max_hours_to_change_lamp :
        tmp = test_fcst.loc[test_fcst.yhat_upper > max_hours_to_change_lamp]
        df_final.at[subset_id, 'date_to_change'] = tmp.iloc[0].ds
        df_final.at[subset_id, 'value_to_change'] = round(tmp.iloc[0].yhat, 0)
        print("Warning! Found date to change UV lamp: {}".format(tmp.iloc[0].ds))

    f, ax = plt.subplots(1)
    f.set_figheight(5)
    f.set_figwidth(15)
    ax.scatter(test.ds, test.y, color='r')
    fig = model.plot(test_fcst, ax=ax)

df_final.to_csv(result_path +'result.csv', mode='a', header=True)
```

Python code. Working hours prediction. Source: [authors research]

```python
#general and weekly trend
if fewDataToPredict == False :
    model.plot_components(test_fcst)
```



Python code. Model components plotting. Source: [authors research]

```python
#values of error metrics
print("mae: {0:.2f} hour(s), mape: {1:.2f}%, predicted value: {2:.2f} hours".\
      format(mae, mape, test_fcst.iloc[-1].trend))
```

```
mae: 156.76 hour(s), mape: 5.97%, predicted value: 2961.91 hours
```

```python
#full info about the predicted period
test_fcst
```

	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_upper	weekly	weekly_
0	2021-12-30	2354.662176	2143.930500	2575.201346	2354.662176	2354.662176	-1.124543	-1.124543	-1.124543	-1.124543	-1.12
1	2021-12-31	2367.313069	2159.146259	2586.836823	2367.313069	2367.313069	0.158624	0.158624	0.158624	0.158624	0.15
2	2022-01-01	2379.963962	2160.944941	2595.170058	2379.963962	2379.963962	-1.303600	-1.303600	-1.303600	-1.303600	-1.30
3	2022-01-02	2392.614855	2183.435079	2602.180019	2392.614854	2392.614855	-2.691117	-2.691117	-2.691117	-2.691117	-2.69
4	2022-01-03	2405.265748	2181.804681	2616.347265	2405.265747	2405.265748	-3.140815	-3.140815	-3.140815	-3.140815	-3.14
5	2022-01-04	2417.916641	2205.379423	2633.723845	2417.916640	2417.916641	2.885454	2.885454	2.885454	2.885454	2.88
6	2022-01-05	2430.567534	2226.060257	2653.183199	2430.567533	2430.567534	5.215998	5.215998	5.215998	5.215998	5.21

Python code. Full information about model results. Source: [authors research]

BIOCAD
Biotechnology Company

РЕКОМЕНДАТЕЛЬНОЕ ПИСЬМО

Елизарьева Наталья и **Титова Диана**, студенты 2-го курса программы магистратуры ВШМ СПбГУ «Бизнес Аналитика и Большие Данные» работали над проектом по прогнозированию проведения техобслуживания (ТО) хроматографов на производстве АО «БИОКАД».

В рамках проекта студентами был проведен разведочный анализ данных и разработаны несколько вариантов моделей для расчета времени наработки уф-ламп. Наталья и Диана продемонстрировали обширные знания в области анализа данных и бизнес-анализа, а также знание библиотек и актуальных методик разработки на языке python, что помогло им успешно справиться с поставленными задачами.

В ходе работы над задачами студенты активно предлагали собственные идеи по реализации, рассматривали различные подходы. Разработанная ими модель была взята в дальнейшую работу в BIOCAD и выступила основой для создания инструмента планирования заказов оборудования и проведения ТО.

Наталья и Диана провели серьезную исследовательскую и практическую работу, зарекомендовали себя в качестве специалистов по аналитике больших данных. Все задачи были выполнены своевременно и качественно.

17.05.2022

Н.А.Хабарова

Junior Data Scientist

BIOCAD

BIOCAD project completion recommendation letter. Source: [authors research]