Fachhochschule Dortmund

University of Applied Sciences and Arts

(Embedded Systems Engineering)

Anomaly detection for Pump using sensor data

Research Project (Thesis)

Author: Dashdamirov, Nijat

Matriculation Number: 7213892

Supervisor: Tabunshchyk, Galyna

Table of Contents

1.Introduction 1

2.Literature review 3

2.1 Pump failure and its types 3

3. Data and methodology 9

3.1 Used sensors 9

3.2 Data acquisition and storage 16

3.2.1 MQTT communication 16

3.2.2 Infineons encryption 17

3.2.3 RDDL 18

3.3 SQL to RDDL 24

3.3.1 Docker 24

4.Analysis and result 27

4.1 Feature extraction 27

4.2 Synthetic data generation 36

4.3 ML implementation 38

4.3.1 One-Class SVM 40

4.3.2 Isolated Forest 44

4.3.3 DBSCAN 47

5.Conclusion 51

5.1 Summary of findings 51

5.2 Limitation of research 52

5.3 Recommendations and future research 52

References 53

Table of figures

Figure 1. Statistics of failure modes in induction motors 4
Figure 2. Functional block diagram of ADXL362........ 10
Figure 3. Functional Block diagram of ADXL356........ 12
Figure 4. Functional Block diagram of BMM150 13
Figure 5. Functional Block diagram of ADT7410........ 14
Figure 6. Functional Block diagram of IM69D130 15
Figure 7. Demonstration of MQTT Broker working principle........ 17
Figure 8. Structure of Infineons internal data lake. 19
Figure 9. Metadata used as metadata for data storage and retrieval. 21
Figure 10. Data base of 3-axis sensor information 23
Figure 11. Yaml file structure used to create containers 24
Figure 12. Graph representation of code data retrieval from MQTT broker........ 25
Figure 13. Logs of Sql connection container creation 26
Figure 14. Logs of Sensor data processing container 26
Figure 15. Workflow of data sensing, processing and storing......... 27
Figure 16. Pie chart indication of proportion of abnormal, normal, synthetic and real data. 37
Figure 17. Line graph of input data for developing model........ 38

Table of Tables

Table 1. Results of different nu variable with 2 different test set......... 42
Table 2. Output of different sets from One-class SVM........ 43
Table 3. Output from Isolated Forest algorithm for different set of indicators and efficiency scores........ 46
Table 4.Output from DBSCAN algorithm for different Nu variable........ 48
Table 5. Result from DBSCAN algorithm with efficiency scores........ 50

1. Introduction

Motors are simple electro-mechanical devices used in industry to obtain mechanical energy from electrical energy. They are robustly constructed engines used not only for general purposes but also in hazardous areas and heavy industrial environments. General purpose applications of induction motors include pumps, conveyors, machine tools, centrifugal machines, elevators and packaging machinery. On the other hand, its applications in hazardous areas are petrochemicals and natural gas facilities. In addition, asynchronous motors are extremely reliable, require low maintenance and provide relatively high efficiency. Moreover, the wide power range of induction motors, from hundreds of watts to megawatts, meets most industrial production needs. But asynchronous motors since they are used extensively in industrial applications, it is possible to encounter mechanical and electrical malfunctions. A motor failure that is not detected at the initial stage can result in disaster and the motor can be seriously damaged. Thus, undetected errors can cause engine failure. This may cause production stoppages. Such downtimes are costly in terms of increased production time, maintenance costs and wasted raw materials. Engine malfunctions are caused by mechanical and electrical parts. Mechanical failures occur due to overloads and sudden load changes, which can lead to bearing failures and rotor bar breakage. On the other hand, electrical faults are usually associated with the power supply. Motors can be driven from fixed frequency sinusoidal power sources or adjustable AC drives. However, asynchronous motors driven by AC drives, are more susceptible to failure. This is because the overvoltage caused by AC drives in stator windings, high-frequency stator current components and bearing-derived currents. Furthermore, motor overvoltage's may occur due to the length of cable connections between the motor and the AC drive. This last effect is due to fluctuating voltage transition [1]. Such electrical effects can also cause short circuits in the stator windings and cause complete motor failure. According to published research, asynchronous motor failures; include bearing failures, stator windings short broken rotor bars and end ring failures. Bearing failures account for approximately two-fifths of all failures. Short circuits in the stator windings represent approximately one third of the detected faults. Broken rotor bars and end ring failures account for approximately ten percent of induction motor failures [2, 3].

The study conducted on 114 motors by the Motor Reliability Working Group [2] and on 6312 motors by the Electric Energy Research Institute [3] shows that bearing, winding and rotor failure groups are the most common failure types.

Various alternative maintenance methods have been used in the industry to prevent serious damage to motors from these fault groups and to prevent unexpected production stoppages. For example; integrity of engines, abnormal vibrations, lubrication a frequent calendar maintenance schedule has been implemented to inspect problems, bearing conditions, stator windings and rotor cage condition. This type of maintenance is done by taking the engine out of service, which means stopping production. Generally, large companies opt for annual maintenance, during which production is stopped for maintenance operations.

In this thesis, it is demonstrated that the vibration analysis method can detect engine malfunctions without the need for damaging tests or plant shutdowns. In particular, the how efficient is it to use vibration data to analyze and which algorithm performs better.

The data here was collected from the air pump that is working in production mode in Infineon. Vibration is kind of indicator that, it reflects any kind of anomaly that occurs in separate part of the motor, including voltage fluctuation and damage in different part pump. The data set, covers 2 month of vibration data of motor, which works only normal mode. No kind of anomaly is present in the data.

Synthetic data generation technique used here to create anomaly cases, to check the accuracy and efficiency of the modals. As it is practically costly and almost impossible to cover all types of anomaly cases, unsupervised learning algorithms implemented. One-class SVM, Isolated Forest and DBSCAN algorithms were chosen, since they are proven to be well-performing especially for anomaly detection. Three different model were developed on the same data, and finally results were compared to conclude which algorithm suits best for our case.

In section 2, main anomaly that happens for motors, together with previous studies conducted on this topic will be covered. Section 3 will mainly focus data collection, processing and synthetic data generation for analysis part. Section 4 will include several tests and different models to detect the anomalies in general. In the last section, will cover the conclusion part together with limitations and recommendations.

2. Literature review

2.1 Pump failure and its types

In the industrial sector, the most used electromechanical energy conversion devices are induction motors. They are omnipresent in our daily lives, powering everything from household appliances like refrigerators to industrial machinery and electric vehicles. The efficiency and versatility of motors make them indispensable in various applications, contributing to automation, robotics, and sustainable development They are the recommended option in many applications because of their affordability, durability, and dependability. However, the most worn components, the bearings, lead to malfunctions in various motor sections because of operating circumstances including humidity, dust, temperature, and lack of lubrication. Techniques for measuring temperature, vibration, bearing oil, and motor current signal analysis are used to identify problems with motors. Traditionally, motor fault detection and classification are carried out by comparing the signal spectrum under both healthy and defective situations.

Unexpected motor failures might still happen in these systems, even with periodic maintenance procedures. This would result in a great deal of downtime as well as significant losses in terms of maintenance costs and income lost. It has been demonstrated that condition-based maintenance (CBM) and predictive maintenance (PdM) are maintenance approaches that can lower unplanned downtime and maintenance costs. [4] Monitoring the condition of electrical machinery can allow early detection of potential catastrophic failures, significantly reducing the cost of maintenance and the risk of unexpected breakdowns. Conditions-based maintenance strategy does not schedule maintenance or machine replacement based on previous records or statistical predictions of machine failure. Instead, it relies on information provided by condition monitoring systems that evaluate the condition of the machine. Therefore, the key to the success of a conditions-based maintenance strategy is to have an accurate condition monitoring and diagnostic tool. Real-time condition monitoring uses measurements taken while a machine is running to determine if any malfunctions are present in the machine.

The overall flow of the predictive maintenance is as follows. Different types of sensors can be used to measure signals to detect these faults. Various signals processing techniques can be applied to signals received from these sensors to extract specific features that are sensitive to the presence of faults. Finally, during the fault detection phase, a decision must be made as to whether a fault exists or not.

Before diving into designing of models initially most common failure types needs to be defined.

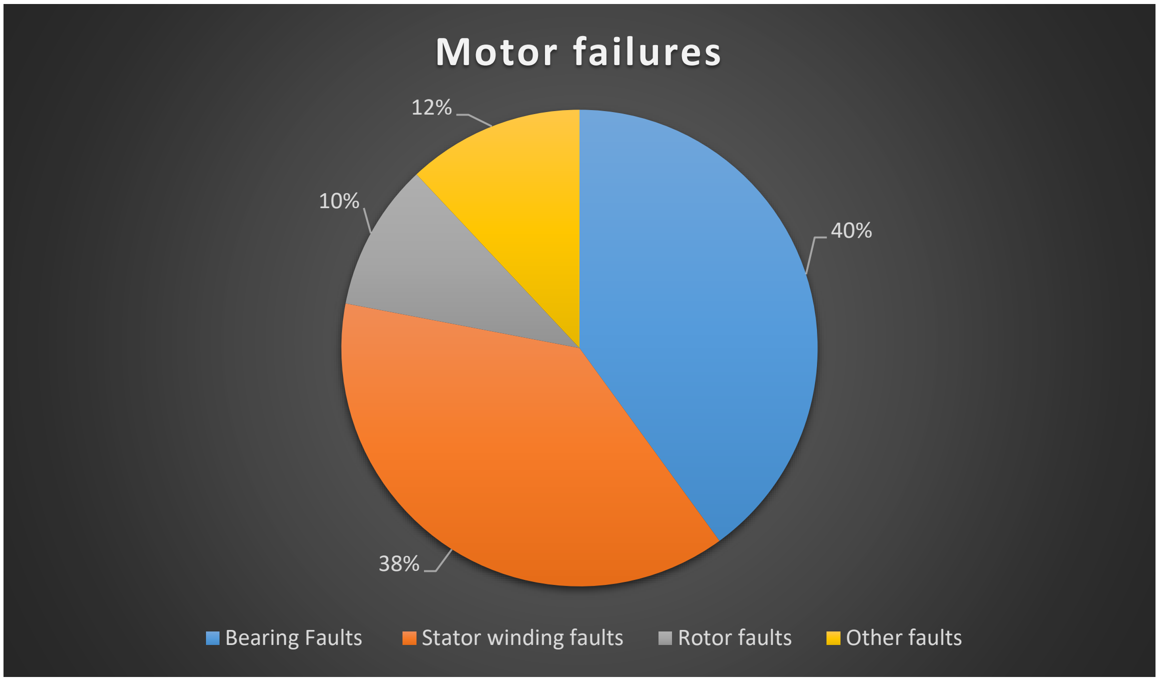


Figure 1. Statistics of failure modes in induction motors

It is evident that over 80% of the reasons for unplanned breakdown in induction motors are rolling-element bearing stator winding failures brought on by insulation deterioration [5].

Bearing faults

Most electrical machines use ball bearings or roller bearings, and they are one of the most common causes of malfunctions. These bearings consist of an inner and outer ring with a series of balls or rolling elements placed in rotating grooves within the rings. Faults in the inner ring, outer ring or rolling elements will produce characteristic frequency components in the vibration measured from the machine and in the signals measured by other sensors. Bearing and housing failure frequencies are defined as a function of bearing geometry and rotational speed [6]. Additionally, bearing failures can cause rotor eccentricity [7].

Stator faults

Approximately 38% of all known asynchronous machine failures fall into this category. The stator coil consists of insulated copper conductor wires placed in the stator grooves. Stator winding faults are usually caused by insulation failure between two adjacent windings in a coil. This is called winding fault or winding short circuit. Induced currents due to short circuits are called short circuit currents. Short circuit current causes excessive winding temperatures and instability in the magnetic field of the machine. An unbalanced magnetic field can cause excessive vibration, which can cause bearing and bearing failures. If left undetected, it starts from the short-circuited windings and spreads to all windings, causing catastrophic stator insulation damage. Insulation faults in the stator windings cause a short circuit between the turns of the stator windings in three-phase asynchronous motors. Initially, the short circuit is only in a few windings, but due to the rapid increase in temperature in that area, the insulation materials of the surrounding windings melt, thus causing the short circuit to spread rapidly.

Rotor faults

Rotor failures constitute approximately 10% of total asynchronous machine failures. The normal failure mechanism is the breakage or cracking of the rotor bars, which can be caused by the thermal or mechanical cycling of the rotor during engine operation. This type of fault causes the formation of double slip frequency sidebands around the supply voltage signal frequency in the current spectrum [8].

$$f_{broken} = f_s(1 \pm 2ks)$$

Here, fs is the supply voltage frequency, s is the slip percentage and k is any number such as 0,1,2…n.

Other faults

Eccentricity failure occurs when the rotor is not placed exactly at the center of the stator, creating an unbalanced air gap between the rotor and stator. This can also be caused by defective bearings or manufacturing defects. The change in air gap negatively affects the magnetic field distribution, which creates a net magnetic force in the direction of the smallest air gap inside the motor. This situation causes mechanical vibration called "unbalanced magnetic pull".

In this thesis, it is demonstrated that the vibration analysis method can detect motor malfunctions without the need for damaging tests or plant shutdowns. Moreover, the presented vibration monitoring method provides the opportunity to continuously monitor motors in operation, thus minimizing human errors that may occur during the detection of motor malfunctions by the relevant operator.

Electrical and mechanical faults occurring in high voltage motors widely used in industry can be detected with advanced fault diagnosis techniques. Different studies on fault diagnosis on these engines are given below.

Mennatallah Amer et al. conducted unsupervised learning approach on predictive maintenance of the motors (Enhancing One-class Support Vector Machines for Unsupervised Anomaly Detection). Main objective of the paper is to focus on different type of Support Vector Machine algorithms. 3 ways of learning are applied: supervised, semi-supervised and unsupervised learning. As SVM algorithms are mostly supervised learning, One-Class SVM specifically is chosen. A one-class SVM projects the data into a higher dimensional space using an implicit transformation function φ that is defined by the kernel. Next, the decision boundary (a hyperplane) separating most of the data from the origin is learned by the algorithm. The decision threshold can only be crossed by a tiny percentage of data points, and those are referred to be outliers [9].

Robust One-class SVMs is based on using the class center as an averaged data source and minimizing the Mean Square Error (MSE) for handling outliers. In comparison to the conventional SVM, the results of the studies demonstrated an improvement in generalization performance and a decrease in the number of support vectors.

Robust one-class SVMs are primarily modified in relation to the slack variables. The slack variables for robust one-class SVMs are proportional to the distance from the centroid. This permits a big slack variable at remote sites from the center. The slack variables are eliminated from the reduction goal since they are fixed.

Unlike strong Robust one-class SVMs, eta one-class SVM method makes use of an explicit outlier suppression strategy. An estimate of a point's normalcy, denoted by the variable η, is introduced to implement this suppression process. Therefore, it would be desirable to set η to zero for an outlying point. The part of the slack variables that will go toward the minimization goal is controlled by this variable. As a result, one-class SVM algorithms performed considerable well in unsupervised learning. Especially, eta one-class SVM outperformed first one in terms of scores.

Tobore Ekwevugbe et al. utilized data mining approach to process and input to model. It is stated that enormous amount of data is stored every day and they are not being processed or analyzed. The experiment consists of four bearings attached to a shaft. To predict failures, only vibration data collected. 4 sensor placed corresponding to each sensor. In order to improve system performance computationally, this study suggests an anomaly detection mechanism that uses optimized characteristics collected from the bearing vibration data. 6 features each bearing are extracted, composed of the standard deviation, variance, skewness, kurtosis, minimum and maximum sensor reading. KNN, SVR and RF algorithms trained here in an unsupervised way. After output it is stated that when compared to SVR and KNN, the RF model has the lowest false positive rate, making it the best machine learning method. SVR is the second-best as it is followed by KNN approach. [10]

R. Ibrahim et al. conducted an anomaly detection for Hydrogenerators by using Autoencoder based vibration signal. For the model train they applied 3 steps of approach. Firstly, healthy vibration signal is collected by wind turbines. Secondly, statistical parameters are extracted. Finally, synthetic data is formed based on the frequency patters that is used to test the model. The ability and sensitivity of the VAE model to identify defects at an early stage have been demonstrated. Ultimately, the model's sensitivity was demonstrated using a series of flawed signals. A collection of in-situ, healthy vibration signals were used as a source for the defective signals, which were manufactured by inserting flaws based on their frequency patterns. The findings demonstrate how potential this method is for diagnosing rotor inter-turn short-circuit problems because of its great sensitivity in identifying defects in their early phases [11].

In their study, Jeffali et al. presented a methodology based on imaging with a thermal camera for fault detection in asynchronous motors, the detection of mechanical faults caused by misaligned couplings that cause heating in the motor body, with infrared technology, and the repercussions of these faults throughout the production chain. For this study, they created a suitable test setup consisting of asynchronous motor, coupling and bearings. As a result of changing the coupling alignment angles on the mechanism; bearings are exposed to more stress and friction, average torque value decreases, efficiency decreases and metal temperature increases in the motor body.

They stated that by monitoring the temperature change of the rotating parts, the remaining useful life of the relevant parts can be estimated [12].

Othman et al. compared the vibration and acoustic diffusion methods used to detect bearing faults that cause catastrophic damage in motors, and evaluated the comparison between time-based domain efficiency and frequency-based domain with the obtained graphs. In the statistical analysis, the variables were RMS, crest factor and kurtosis for the time-based domain are utilized. They also applied normal and envelopment techniques and "Hilbert" transformation for the frequency domain. Based on their results, they showed that vibration and acoustic emission signals are effective for detecting motor bearing failure in both time-based and frequency-based domains [13].

In their study, Hulugappa et al. compared vibration, stator current, acoustic emission, shock impact and surface analysis measurements by applying different loads and speeds on the damaged bearing in a motor. The test setup, prepared from an asynchronous motor, belt pulley and load system, was created to observe the condition of the damaged bearing on the motor drive side. Piezoelectric accelerometer and FFT analyzer were used for vibration measurement, current collector and FFT analyzer for stator current measurement, transducer, amplifier and some filters for acoustic emission measurement, and hand-held impact meters for shock pulse measurement. According to the results, the effectiveness of the measurement methods in detecting damaged bearing failure was expressed as acoustic emission, shock pulse, vibration and stator current measurements, respectively. However, they reported that stator current measurement is advantageous as a technique that requires minimum equipment [14].

Nik Dennler et al. did a research about detecting anomalies by using vibration data in an online way. Their main prospective is to achieve type of model that it is applicable to smaller scale operations such as such as autonomous cars, drones. In starting phase, they implemented frequency decomposition on acoustic waves. Set of linear filters are organized tonotopically, each tuned to a specific center frequency. Since they capture many characteristics of the biological cochlear functioning using analog and digital event-based circuits, they serve as a crucial building component for low-power auditory processing. Then, filtered signals are encoded into asynchronous spike trains to benefit from power advantages of spike-based signal processing. Finally processed signal input to SNN. As a result, neuromorphic pipeline that uses vibration pattern records to identify machine abnormalities. Fault detection timings demonstrated that abnormalities are reported in a robust and accurate way [15].

Clemens Heistracher et al. used transfer learning strategies for anomaly detection. The research was carried out by collaboration of Siemens and Austrian Institute of Technology. As it is costly and almost implausible to cover all kind of anomalies, they proposed to have a model trained and use it for different kind of faultiness. They considered that, it is not the type of algorithm but the data that matters in terms of efficiency of model. The goal of this study is to develop and put into practice a method that minimizes the quantity of data needed to train anomaly detection models. This should lessen the total effort necessary to install IoT sensors across similar industrial assets. Three types of learning implemented: No transfer learning, Transfer learning with labeled data and Transfer learning with unlabeled data. After feature extraction, all types of learning are tested. Tests show that, in comparison to non-transfer learning, pre-training using labeled data from other assets enhances the F1-score. If there is just unlabeled data available, nevertheless it can make progress using a small number of training samples from the target asset. It seems that the impact of an embedding can be mitigated by well labeled examples. Overall, the study suggests that models trained on vibration sensor data should be transferred across industrial assets of the same kind using both supervised and unstructured approaches [16].

3. Data and methodology

3.1 Used sensors

ADXL362

The ADXL362 distinguishes out as an ultralow power, 3-axis MEMS accelerometer with a strong focus on energy saving. The sensor is an excellent solution for applications where power saving is critical, with a power consumption of less than 2 A at a 100 Hz output data rate and an incredibly low 270 nA in motion-triggered wake-up mode.

The ADXL362 is notable by its strategy to achieve low power consumption. Unlike accelerometers that use power duty cycling, this sensor prevents aliasing input signals by sampling the whole sensor bandwidth at all data rates. This provides precise and dependable data while preserving power efficiency.

The sensor consistently provides 12-bit output resolution, and for more streamlined data transfers, an 8-bit formatted data option is available when lower resolution suffices. In our case 12-bit output used. With measurement ranges of ±2 g, ±4 g, and ±8 g, the ADXL362 offers flexibility to accommodate various application needs. The high resolution of 1 mg/LSB on the ±2 g range enables precise motion detection, suitable for a wide range of scenarios.

In pursuit of reduced noise levels, the ADXL362 provides two lower noise modes, achieving as low as 175 µg/√Hz typical, with only a minimal increase in supply current. This versatility makes it adaptable to applications demanding enhanced noise performance.

The ADXL362 has capabilities for full system-level power saving in addition to its ultralow power consumption. These include a deep multimode output FIFO, a built-in micro power temperature sensor, and multiple activity detection modes, such as configurable threshold sleep and wake-up operations, capable of operating at as low as 270 nA at a measurement rate of 6 Hz (approximate). When activity is detected, a pin output enables for direct control of an external switch, allowing further customization choices.

The ADXL362 operates within a wide supply range of 1.6 V to 3.5 V and can interface with a host operating on a separate, lower supply voltage if needed. The sensor's compact size, with dimensions of 3 mm × 3.25 mm × 1.06 mm, further enhances its suitability for integration into space-constrained designs [17].

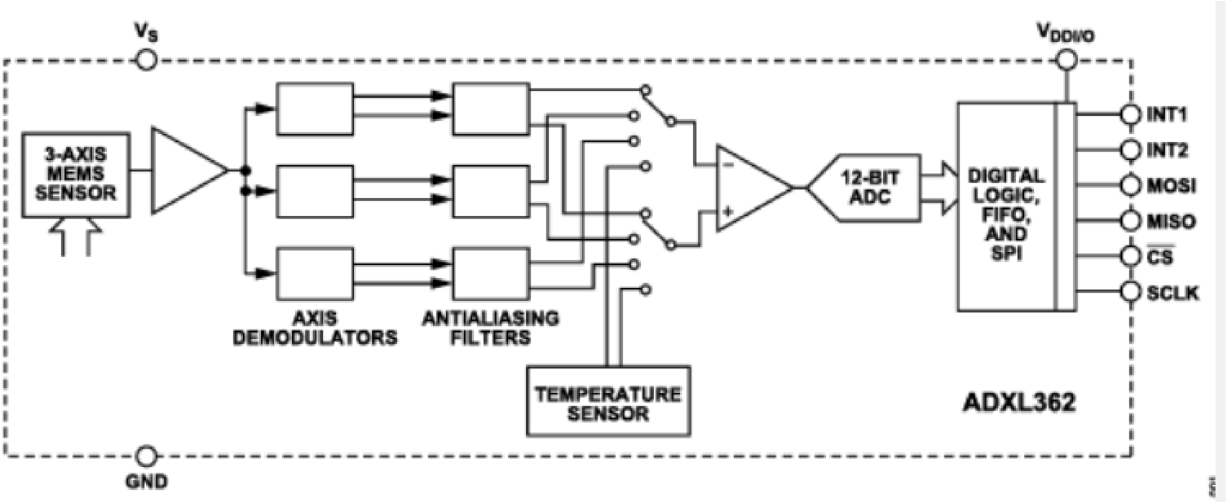


Figure 2. Functional block diagram of ADXL362

ADXL356

The Analog Devices ADXL356 and ADXL357 offer a potent combination of low noise density, low 0 g offset drift, and low power consumption, establishing them as high-performance 3-axis accelerometers with customizable measurement ranges. These sensors are suitable for a variety of precision-required applications, and their configurable measurement ranges provide adaptability.

The ADXL356B, which supports ±10 g and ±20 g ranges, is designed for applications that require a moderate range of acceleration. Meanwhile, the ADXL356C adds versatility by supporting ±10 g and ±40 g ranges, allowing it to accommodate scenarios with varied acceleration magnitudes. The ADXL357 expands the variety of applications it may serve by supporting ±10 g, ±20 g, and ±40 g ranges.

An important characteristic of the ADXL356 and ADXL357 is its long-term stability, minimum offset drift with temperature, and industry-leading noise performance. These features help the sensors give accurate measurements with low calibration needs, which makes them useful in applications where precision is crucial.

Low drift, low noise, and low power consumption of the ADXL357 become very useful in highly vibration-prone settings. This sensor demonstrates its strong performance in difficult circumstances by measuring tilt accurately even when there are strong vibrations present.

The ADXL356 is best suited for condition-based monitoring and other vibration sensing applications due to its low noise characteristics at higher frequencies. As a result, it's a useful instrument for applications requiring sensitivity to subtle variations in acceleration, offering information on the operational state and health of the equipment.

The ADXL357 has multifunction pin designations that may be precisely referred for their appropriate tasks within the restricted I2C interface or serial peripheral interface (SPI) for simplicity of usage and integration. This improves the user experience overall by simplifies the design and execution processes [18].

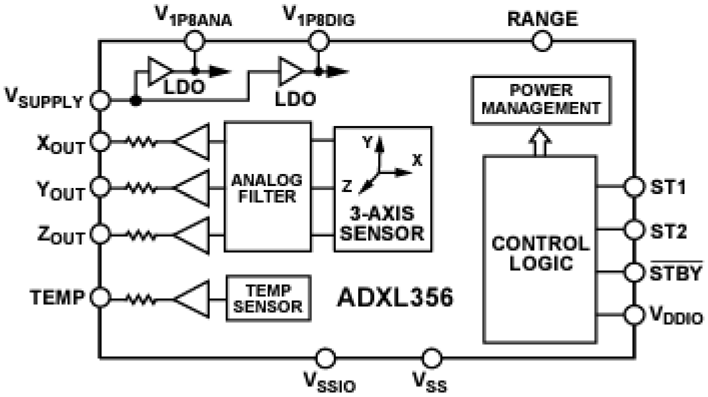


Figure 3. Functional Block diagram of ADXL356

BMM150

The BMM150 is a stand-alone geomagnetic sensor designed for consumer market applications that need precision magnetic field measurements in three perpendicular axes. The BMM150, which leverages Bosch's unique FlipCore technology, provides precisely tailored performance that is ideally aligned with the hard needs of 3-axis mobile applications. This sensor finds its niche in applications such as electronic compasses, navigation systems, and motors, where accurate geomagnetic data is paramount.

An assessment circuitry (ASIC) at the core of the BMM150 effectively translates the sensor's geomagnetic output into digital findings. These findings may be easily accessible using industry-standard digital interfaces such as SPI (Serial Peripheral Interface) and I2C (Inter-Integrated Circuit), improving compatibility and simplicity of integration with a wide range of devices.

The BMM150's box and interfaces were designed with adaptability in mind, catering to a wide range of hardware needs. Its ultra-small footprint and compact design make it ideal for mobile applications. With dimensions as small as 1.56 x 1.56 x 0.6 mm3, the wafer level chip scale package (WLCSP) guarantees considerable flexibility in printed circuit board (PCB) positioning, maximizing space consumption in tiny devices.

The BMM150's ultra-low voltage functioning is one of its most notable characteristics. The sensor is not only power-efficient, but also adaptable to a variety of power supply designs, with a voltage range of 1.62V to 3.6V for VDD and 1.2V to 3.6V for VDDIO. The BMM150 is also programmable, allowing developers to improve functionality, performance, and power consumption according on the individual needs of their applications. A programmable interrupt engine gives another level of design freedom.

Because the BMM150 excels at measuring the three axes of the terrestrial magnetic field, it is an essential component in a wide range of consumer electronics. Cell phones, mobile gadgets, computer peripherals, man-machine interfaces, virtual reality features, and gaming controllers are all examples of its usage. The BMM150 contributes to the flawless operation of these devices by giving precise and reliable geomagnetic data, improving user experience, and increasing the possibilities of novel technology [19].

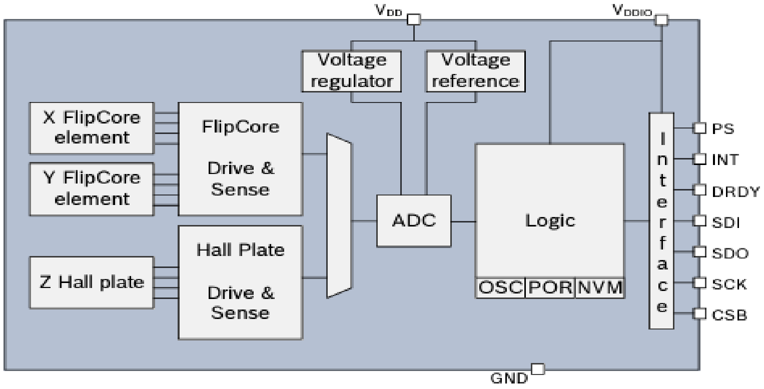


Figure 4. Functional Block diagram of BMM150

ADT7410

The ADT7410 is the pinnacle of digital temperature sensing, combining great precision in a small SOIC device. This sensor has a band gap temperature reference and a 13-bit ADC, allowing for accurate temperature monitoring and digitization with an outstanding 0.0625°C resolution.

The ADT7410's devotion to precision is demonstrated by the preset ADC resolution of 13 bits (0.0625°C), but its adaptability extends much further. Users may increase resolution to 16 bits (0.0078°C) by simply changing Bit 7 in the configuration register (Register Address 0x03). This versatility is useful in situations requiring finer temperature details.

The ADT7410 is intended for dependability in a wide range of voltage settings, operating effortlessly over supply voltages ranging from 2.7 V to 5.5 V. The sensor finds a compromise between accuracy and power efficiency at a typical average supply current of 210 A while running at 3.3 V. Furthermore, the ADT7410 has a shutdown mode that effectively powers down the device while lowering current usage to a paltry 2 A.

The ADT7410 is designed to withstand tough climatic conditions, with a temperature range extending from -55°C to +150°C. Because of its wide temperature range, it is appropriate for use in both extreme cold and extreme hot conditions.

Pins A0 and A1 are dedicated to address selection for improved configurability, giving the ADT7410 four different I2C addresses. This functionality allows for the easy integration of several I2C devices into diverse systems.

When the temperature exceeds a user-programmable critical temperature limit, the CT (Critical Temperature) pin activates as an open-drain output. The critical temperature limit is set to 147°C by default, providing an extra layer of safety for applications that are sensitive to high temperatures. When the temperature reaches a preset limit, the INT (Interrupt) pin, which is also an open-drain output, becomes active. Both the INT and CT pins may work in either comparator or interrupt mode, allowing them to adapt to diverse system requirements [20].

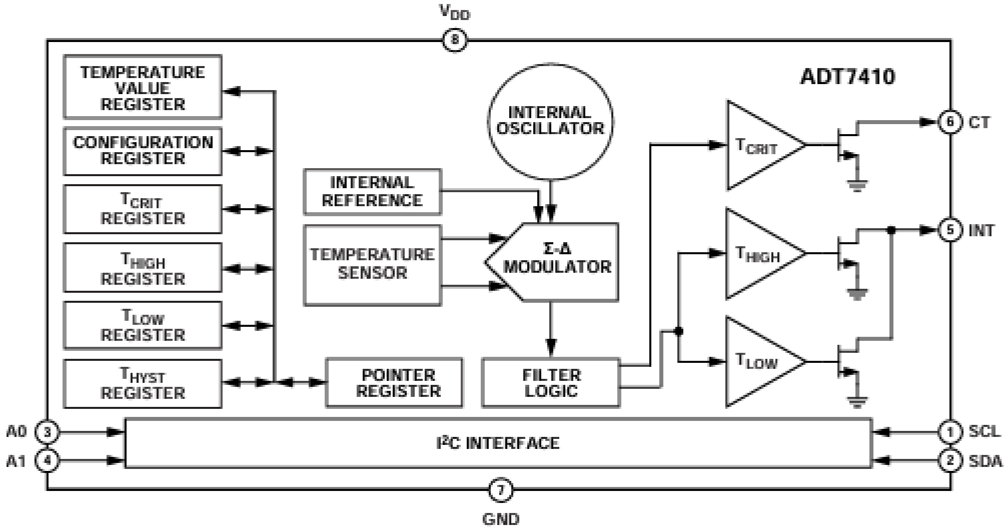


Figure 5. Functional Block diagram of ADT7410

IM69D130

The Infineon IM69D130 is a precision-engineered microphone designed for applications requiring outstanding audio performance. This microphone uses Infineon's Dual Backplate MEMS technology to achieve low self-noise, a broad dynamic range, minimum distortions, and a high acoustic overload point. This new technique is based on a tiny symmetrical microphone architecture, similar to studio condenser microphone arrangements. The end product is a very linear output signal with a dynamic range of 105dB.

Notably, even in difficult settings, the IM69D130 retains remarkable performance. Even at sound pressure levels of 128dB, the microphone's distortion remained below 1%. This level of accuracy makes it ideal for applications requiring high audio quality.

The flat frequency response of the microphone, with a low-frequency roll-off at 28Hz, along with rigorous production tolerances, guarantees close phase matching between microphones. This feature is critical for applications using several microphones in an array, since it contributes to the microphone's smooth integration into complicated audio systems.

The IM69D130's low equivalent noise floor of 25dB distinguishes it as a high-performance component in the audio signal chain. Because of the low noise floor, the microphone does not hinder the performance of voice recognition algorithms, allowing for maximum accuracy and responsiveness.

The IM69D130 is powered by a digital microphone ASIC that includes an exceptionally low-noise preamplifier and a high-performance sigma-delta ADC (Analog-to-Digital Converter). This combination allows for accurate signal conversion while reducing noise interference. With adjustable power modes, the microphone provides users with the ability to customize current consumption to individual needs.

Each IM69D130 microphone is calibrated using a sophisticated Infineon algorithm, resulting in sensitivity tolerances of 1dB. Furthermore, the phase response is well matched (2°) between microphones, which supports beamforming applications that need accurate phase alignment [21].

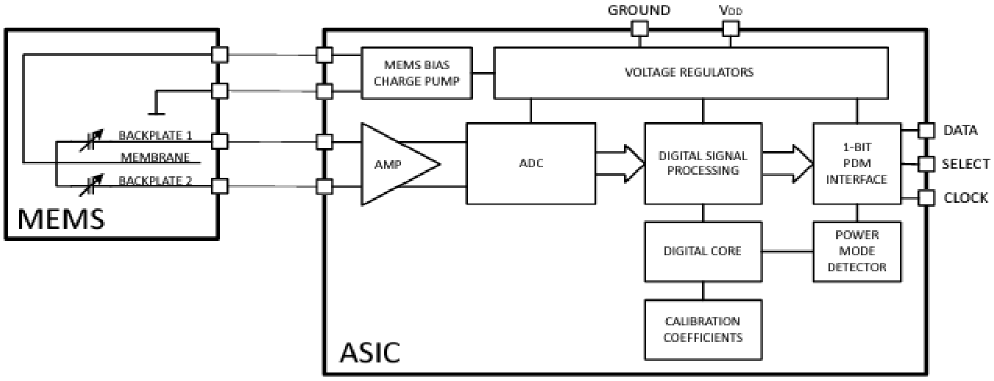


Figure 6. Functional Block diagram of IM69D130

3.2 Data acquisition and storage

3.2.1 MQTT communication

MQTT, or Message Queuing Telemetry Transport, is a lightweight and open-source messaging protocol designed primarily for the efficient communication of tiny sensors and mobile devices, especially in contexts with high latency or unreliable networks. MQTT's communication paradigm is built on three important components: publishers, subscribers, and brokers. The MQTT broker appears as a critical component in this environment, acting as a server that orchestrates communication between MQTT clients.

The MQTT broker performs numerous critical functions in the publish/subscribe paradigm. It accepts messages from clients, routes them to subscribing clients based on predefined themes, and maintains the overall flow of communication between publishers and subscribers. Because of this design, communication may be simplified and decoupled, with the broker handling all of the intricate details of publisher-subscriber interactions [22].

Managing connectivity by managing multiple client connections at once, enabling topic-based messaging to route messages to interested subscribers, supporting various Quality of Service (QoS) levels for dependable message delivery, managing sessions to store and deliver messages for clients who are momentarily disconnected, and putting security measures like authentication and authorization in place are some of the key functions of a MQTT broker.

There are several MQTT broker implementations available, ranging from open-source solutions like Mosquitto, which are known for their lightweight nature, to enterprise-ready options like HiveMQ and EMQ X Broker, which offer scalability and robust features suitable for large-scale deployments, particularly in the context of the Internet of Things (IoT). MQTT brokers play a critical role in the IoT ecosystem, providing efficient, real-time communication between IoT devices in areas with limited bandwidth and unpredictable connectivity. These brokers areintended to be scalable, allowing clusters to be deployed to handle enormous IoT installations, with redundancy features assuring ongoing operation even in the event of probable node failures.

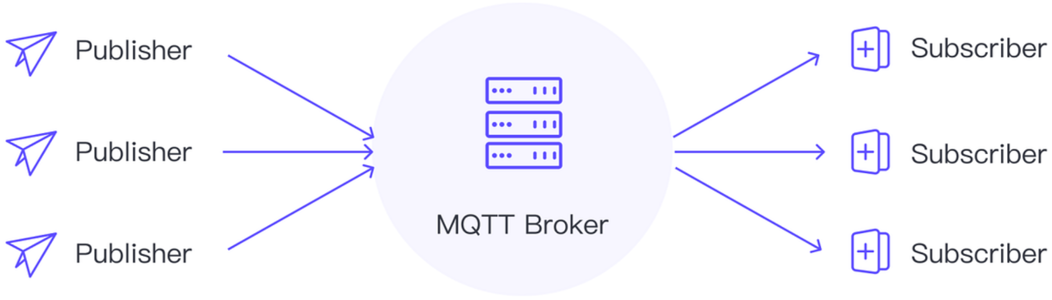


Figure 7. Demonstration of MQTT Broker working principle

In this project Mqtt broker acts as messaging protocol between the sensors and server. For the sake of safeness, broker itself were developed by the company itself. Publisher are icomoxbox while subscriber is the script that is developed by the team. Data is sent by using encryption methods.

3.2.2 Infineons encryption

Infineon uses its own encryption. There are multiple modes while sending the message. Message format is in string and it is 64-bit hexadecimal format. First 4 signs define type of the message. While starting the project to test if it is working correct or not “Hello” type message is sent. If it is successfully being accepted by the subscriber, it means the connectivity established successfully.

Second one is “Reset”. It can be imagined as reset pin in the devices. There can be cases where, broker is overloaded or there a problem in connection, that message should be sent. In that case connection will be reset. As it is mentioned earlier, broker has feature of storing messages. In case of “Reset”, previous messages will be deleted, thus that needs to be used very carefully.

Setting and Getting config credentials. That is really important part of the project. Once the connection is started, it is aimed to ensure message delivery. To guarantee safety, there should be some config information. Setter and getter methods, help us to retrieve and determine the config files.

“Info” helps us to get information about the broker, brief description, name and messages it contains.

Finally, “Report” comes into play. It is the main point where the message is being sent. Report contains information from the all the sensors. With reference to the sensors phase, message differs according to each sensor. It depends on the frequency and axis of the signal. Additionally, message contains, BoardType, MCUSerial, Name and etc. Once message encrypted successfully it is ready to deliver.

3.2.3 RDDL

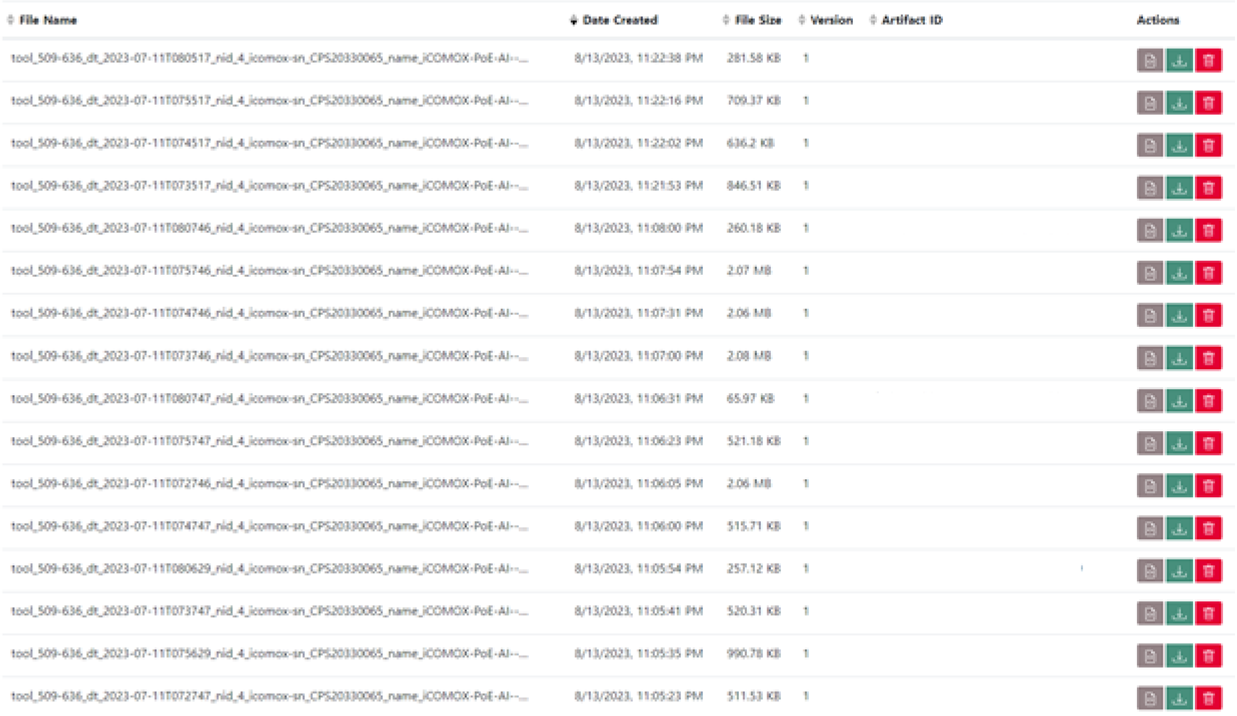
A data lake emerges as a disruptive solution in modern data management, functioning as a centralized and scalable repository meant to store huge amounts of raw, unstructured, and heterogeneous data in its natural format until the need for analysis arises. Unlike typical databases, data lakes abandon the confines of predetermined schemas and inflexible hierarchies in favor of keeping data in its rawest and natural form. This adaptability enables enterprises to handle and create value from an ever-expanding range of data kinds, including text, photos, videos, log files, and others, without the need for costly preparation.

A data lake's primary feature is its capacity to serve as a repository for both structured and unstructured data, offering an integrated platform for storage that promotes a more complete and all-encompassing view of the data landscape of an organization. A data lake allows for the on-demand exploration and analysis of data by data scientists, analysts, and other stakeholders by keeping the data in its unprocessed state. This allows for a more flexible and iterative approach to data analytics.

Data lakes are essential for dismantling the conventional data that impede internal cooperation and insight-gathering in enterprises. Data lakes facilitate cross-functional cooperation by consolidating disparate data sources into a single repository, enabling many teams and departments to access and utilize a common data set. This democratization of data access enables people within the business to gain insights from a wide range of data sources, supporting creativity and informed decision-making.

The potential of data lakes to enable advanced analytics, artificial intelligence, and machine learning programs is one of its primary advantages. The amount and diversity of data housed in a data lake make it an ideal environment for training machine learning models, identifying important patterns, and delving deeper into complicated datasets. This establishes the data lake as a vital infrastructure for enterprises seeking actionable insight from their data.

A data lake's strategic deployment becomes essential to contemporary data architecture as more and more enterprises realize the value of their data assets. Nevertheless, to avoid turning a data lake into a "data swamp" where data relevance and quality might eventually deteriorate, careful design, strong governance, and efficient administration are essential to its success.



File Name	Date Created	File Size	Version	Artifact ID	Actions
tool_509-636_dt_2023-07-11T080517_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:22:38 PM	281.58 KB	1		
tool_509-636_dt_2023-07-11T075517_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:22:16 PM	709.37 KB	1		
tool_509-636_dt_2023-07-11T074517_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:22:02 PM	636.2 KB	1		
tool_509-636_dt_2023-07-11T073517_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:21:53 PM	846.51 KB	1		
tool_509-636_dt_2023-07-11T080746_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:08:00 PM	260.18 KB	1		
tool_509-636_dt_2023-07-11T075746_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:07:54 PM	2.07 MB	1		
tool_509-636_dt_2023-07-11T074746_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:07:31 PM	2.06 MB	1		
tool_509-636_dt_2023-07-11T073746_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:07:00 PM	2.08 MB	1		
tool_509-636_dt_2023-07-11T080747_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:06:31 PM	65.97 KB	1		
tool_509-636_dt_2023-07-11T075747_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:06:23 PM	521.18 KB	1		
tool_509-636_dt_2023-07-11T072746_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:06:05 PM	2.06 MB	1		
tool_509-636_dt_2023-07-11T074747_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:06:00 PM	515.71 KB	1		
tool_509-636_dt_2023-07-11T080629_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:05:54 PM	257.12 KB	1		
tool_509-636_dt_2023-07-11T073747_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:05:41 PM	520.31 KB	1		
tool_509-636_dt_2023-07-11T075629_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:05:35 PM	990.78 KB	1		
tool_509-636_dt_2023-07-11T072747_nid_4_icomox-sn_CPS20330065_name_iCOMOX-PoE-AI--...	8/13/2023, 11:05:23 PM	511.53 KB	1		

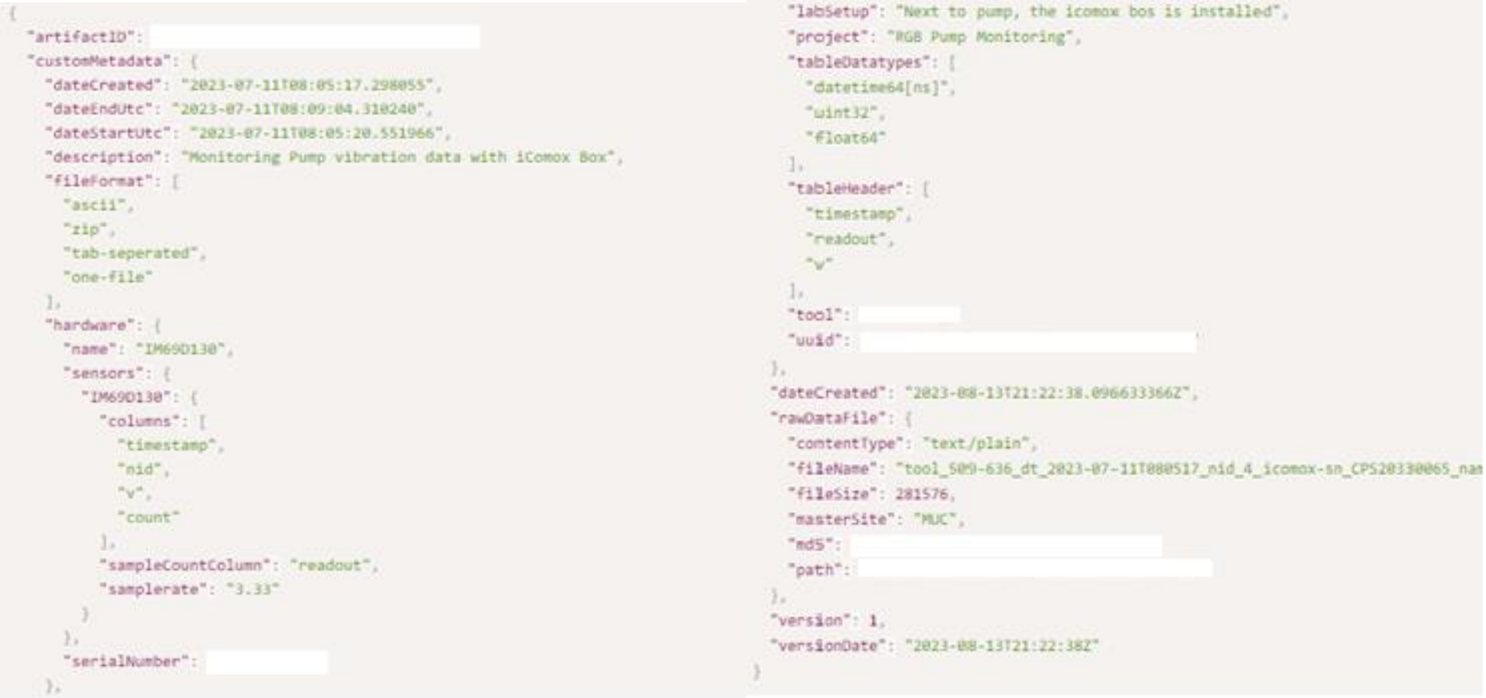
Figure 8. Structure of Infineons internal data lake.

The ability of the data lake to enable flexible retrieval mechanisms, allowing users to access and analyze data on-demand, is crucial. Data lakes provide a variety of techniques for retrieving and analyzing data, allowing users to get valuable insights. Techniques include:

- Schema-on-Read Approach: Data lakes employ a schema-on-read technique, as opposed to typical databases, which use a schema-on-write approach. This means that the data's structure and schema are decided during analysis rather than at the time of input. This method provides agility by allowing users to interpret and arrange data based on their individual analytical requirements.
- Query Languages: Supporting query languages like SQL, data lakes enable users to interactively query and analyze the data using well-known languages. This makes data retrieval and analysis easier and more accessible to a wider group of users inside the company.

- Metadata Management: Metadata plays a crucial role in data lakes by providing information about the data stored within the lake. Sophisticated methods of managing metadata, such as categorization and tagging, allow users to quickly find and access pertinent information. By utilizing metadata, this method improves data governance and guarantees that users can locate the appropriate data when they need it.
- Data Indexing: Data lakes frequently use indexing strategies to improve retrieval performance. Data extraction times may be considerably decreased by building indexes on certain qualities or fields, allowing for faster access to relevant data.
- Integration with Analytics Tools: Data lakes interface smoothly with a wide range of analytics and visualization tools. This interface allows users to access data straight from the lake into their preferred analytics environment, resulting in a simpler and more effective analysis process.
- Machine Learning for Predictive Retrieval: Using machine learning techniques within data lakes enables predictive data retrieval. These algorithms may examine previous usage patterns, user behaviors, and data access frequencies to anticipate which datasets will be accessed, hence maximizing data storage and retrieval efficiency.

There multiple tools online as a data lake, but as a safety rule, Infineon developed its own data lake structure. As it is mentioned there are plenty of ways to retrieve data, while we utilize Metadata Management. Metadata in our case also serves as a data storage where we easily store redundant data.



```
{
  "artifactID": [redacted]
  "customMetadata": {
    "dateCreated": "2023-07-11T08:05:17.298055",
    "dateEndUtc": "2023-07-11T08:09:04.310240",
    "dateStartUtc": "2023-07-11T08:05:20.551966",
    "description": "Monitoring Pump vibration data with iComox Box",
    "fileFormat": [
      "ascii",
      "zip",
      "tab-seperated",
      "one-file"
    ],
    "hardware": {
      "name": "IM69D130",
      "sensors": {
        "IM69D130": {
          "columns": [
            "timestamp",
            "nid",
            "v",
            "count"
          ],
          "sampleCountColumn": "readout",
          "samplerate": "3.33"
        }
      },
      "serialNumber": [redacted]
    },
    "labSetup": "Next to pump, the icomox bos is installed",
    "project": "RGB Pump Monitoring",
    "tableDatatypes": [
      "datetime64[ns]",
      "uint32",
      "float64"
    ],
    "tableHeader": [
      "timestamp",
      "readout",
      "v"
    ],
    "tool": [redacted]
    "uuid": [redacted]
  },
  "dateCreated": "2023-08-13T21:22:38.0966333662",
  "rawDataFile": {
    "contentType": "text/plain",
    "fileName": "tool_509-636_dt_2023-07-11T080517_nid_4_icomox-sn_CPS20330065_nam
    "fileSize": 281576,
    "masterSite": "MUC",
    "md5": [redacted]
    "path": [redacted]
  },
  "version": 1,
  "versionDate": "2023-08-13T21:22:38Z"
}
```

Figure 9. Metadata used as metadata for data storage and retrieval.

Above picture is a live demonstration of the metadata is created before uploading to data lake. The method that Infineons software work with is metadata retrieval. Basically, metadata helps us to improve the ability to search, meanwhile it contains detailed information about the file itself. At Infineon the file format was chosen to be in .tab format as it is easily readable and editable, compatible with all the operation systems. As enormous amount of data is stored inside of one file, the file size needs to be reduced. That's why, before uploading it to data lake all the files are zipped inside of one file.

Every file inside of data lake has unique hash where it is used to retrieve it from the server. In order to make data range search "dateCreated" and "dateEndUtc" is formed. Then in "fileFormat" field gives information about the file itself. What is the alphabet code, file format, how columns are separated from each other and how many files the zip file contains?

In "hardware" section, we can get information about the what is the name of the sensor, name of columns, sample count column and sample rate. Serial number is name of the box which is used here. Also we can use sensor name to do search here.

“tableDataTypes” contains information about the variables that was mentioned in hardware section. We need variable types to cast them because by default our code does not have ability to understand variable format.

“tableHeader” is the headers where we can encounter them after opening the file. It is placed on the first column of the file, because of this while reading it, first column should be skipped.

“uuid” is assigned to each project, meaning that it is unique to project itself. It helps us to distinguish between project files. For example, if we do a search on uuid variable it is going to show us files that only belongs to specific project.

“filename” is the name of the uploaded zip file. The file names are designed in a way that it contains information that is redundant to put to .tab file. For example, each box has its own id while nid appears on the name. It means that this file belongs to icomoxbox with id 4. Then tool name, serial number, name of the project also stored on the name.

“md5” section is an encrypted version of the file name. MD5, or Message Digest Algorithm 5. Its principal function is to create a 128-bit hash value, which is often used for checksums and digital signatures. MD5, which was initially acclaimed for its speed and simplicity, has fallen out of favor for cryptographic applications due to flaws that allow collision attacks. Different inputs can give the same hash result in such assaults, jeopardizing data integrity. The technique divides input data into 512-bit blocks and applies different bitwise operations and modular additions to get the final hash. Despite being obsolete in favor of more secure hash functions such as SHA-256, MD5 is still used in non-security-critical situations such as file integrity verification and older systems. If we take a close look at the path that encrypted version gives us directive where we can have access to file.

Finally, there are some variables changes over project. This also helps us to have information about the project of the file. “description” delivers information about the characteristics of the project. By reading that anyone can have an idea about the project.

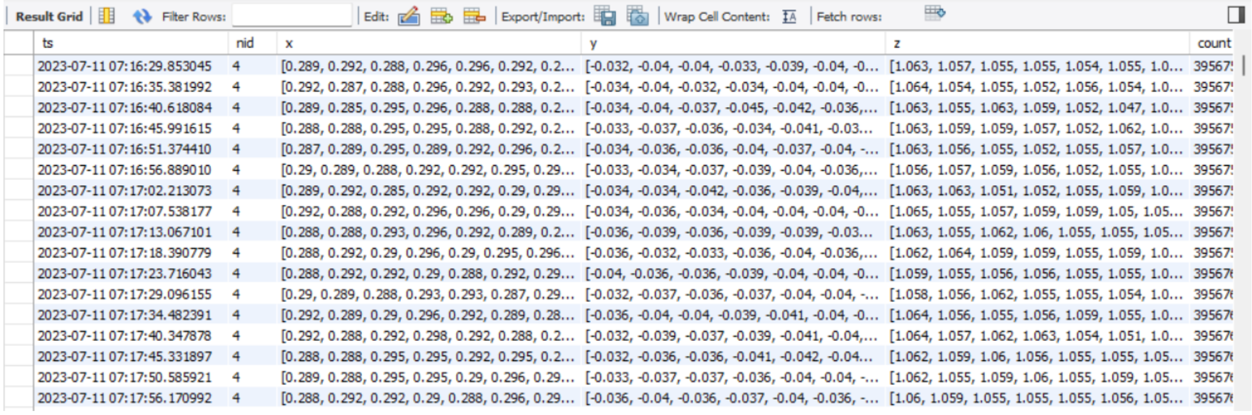
Next, “labSetup” variable embraces information about the setup of the project. It can be different information namely, what kind of sensors are utilized, where they are placed, what they are sensing and etc. Any kind of report that is suitable for project can be present.

SQL

SQL, or Structured Query Language, is a robust domain-specific programming language used to manage and manipulate relational databases. SQL, which was created in the 1970s, is a standard method of connecting with and retrieving information from databases. Its key capabilities include creating and changing database structures, querying data, and conducting operations such as record insertion, deletion, and updating. SQL is well-known for its declarative syntax, which allows users to describe what data they want without having to enumerate the processes required to retrieve it. SQL is now accessible to both beginner and professional database managers. Its flexibility goes beyond simple CRUD (Create, Read, Update, Delete) activities, allowing for the construction of complicated transactions, data integrity requirements, and data analytic tasks. SQL is a vital tool for efficiently maintaining and obtaining valuable insights from relational databases, whether utilized in web development, corporate intelligence, or data science [23].

In this project data is also stored inside of SQL database. There can be questions which data is stored inside of data lake, why do we need to do additional operation which consumes money and time.

As I talked about it earlier, data lake is placed on a separate server and we connect it via internet. Data lake sometimes needs to be optimized and should go through maintenance and it will not be available for use. Furthermore, there can be cases, which internet connection is problematic and no data can be sent to server. To secure system from such kind of cases, SQL database storage acts as a buffer. Data received from the server initially is put to local database to avoid data loss. Then the script will start to run to extract data from database, process it and upload it to data lake.



ts	nid	x	y	z	count
2023-07-11 07:16:29.853045	4	[0.289, 0.292, 0.288, 0.296, 0.296, 0.292, 0.2...	[-0.032, -0.04, -0.04, -0.033, -0.039, -0.04, -0...	[1.063, 1.057, 1.055, 1.055, 1.054, 1.055, 1.0...	39567
2023-07-11 07:16:35.381992	4	[0.292, 0.287, 0.288, 0.296, 0.292, 0.293, 0.2...	[-0.034, -0.04, -0.032, -0.034, -0.04, -0.04, -0...	[1.064, 1.054, 1.055, 1.052, 1.056, 1.054, 1.0...	39567
2023-07-11 07:16:40.618084	4	[0.289, 0.285, 0.295, 0.296, 0.288, 0.288, 0.2...	[-0.034, -0.04, -0.037, -0.045, -0.042, -0.036,...	[1.063, 1.055, 1.063, 1.059, 1.052, 1.047, 1.0...	39567
2023-07-11 07:16:45.991615	4	[0.288, 0.288, 0.295, 0.295, 0.288, 0.292, 0.2...	[-0.033, -0.037, -0.036, -0.034, -0.041, -0.03...	[1.063, 1.059, 1.059, 1.057, 1.052, 1.062, 1.0...	39567
2023-07-11 07:16:51.374410	4	[0.287, 0.289, 0.295, 0.289, 0.292, 0.296, 0.2...	[-0.034, -0.036, -0.036, -0.04, -0.037, -0.04, -...	[1.063, 1.056, 1.055, 1.052, 1.055, 1.057, 1.0...	39567
2023-07-11 07:16:56.889010	4	[0.29, 0.289, 0.288, 0.292, 0.292, 0.295, 0.29...	[-0.033, -0.034, -0.037, -0.039, -0.04, -0.036,...	[1.056, 1.057, 1.059, 1.056, 1.052, 1.055, 1.0...	39567
2023-07-11 07:17:02.213073	4	[0.289, 0.292, 0.285, 0.292, 0.292, 0.29, 0.29...	[-0.034, -0.034, -0.042, -0.036, -0.039, -0.04,...	[1.063, 1.063, 1.051, 1.052, 1.055, 1.059, 1.0...	39567
2023-07-11 07:17:07.538177	4	[0.292, 0.288, 0.292, 0.296, 0.296, 0.29, 0.29...	[-0.034, -0.036, -0.034, -0.04, -0.04, -0.04, -0...	[1.065, 1.055, 1.057, 1.059, 1.059, 1.05, 1.05...	39567
2023-07-11 07:17:13.067101	4	[0.288, 0.288, 0.293, 0.296, 0.292, 0.289, 0.2...	[-0.036, -0.039, -0.036, -0.039, -0.039, -0.03...	[1.063, 1.055, 1.062, 1.06, 1.055, 1.055, 1.05...	39567
2023-07-11 07:17:18.390779	4	[0.288, 0.292, 0.29, 0.296, 0.29, 0.295, 0.296...	[-0.036, -0.032, -0.033, -0.036, -0.04, -0.036,...	[1.062, 1.064, 1.059, 1.059, 1.055, 1.059, 1.0...	39567
2023-07-11 07:17:23.716043	4	[0.288, 0.292, 0.292, 0.29, 0.288, 0.292, 0.29...	[-0.04, -0.036, -0.036, -0.039, -0.04, -0.04, -0...	[1.059, 1.055, 1.056, 1.056, 1.055, 1.055, 1.0...	39567
2023-07-11 07:17:29.096155	4	[0.29, 0.289, 0.288, 0.293, 0.293, 0.287, 0.29...	[-0.032, -0.037, -0.036, -0.037, -0.04, -0.04, -...	[1.058, 1.056, 1.062, 1.055, 1.055, 1.054, 1.0...	39567
2023-07-11 07:17:34.482391	4	[0.292, 0.289, 0.29, 0.296, 0.292, 0.289, 0.28...	[-0.036, -0.04, -0.04, -0.039, -0.041, -0.04, -0...	[1.064, 1.056, 1.055, 1.056, 1.059, 1.055, 1.0...	39567
2023-07-11 07:17:40.347878	4	[0.292, 0.288, 0.292, 0.298, 0.292, 0.288, 0.2...	[-0.032, -0.039, -0.037, -0.039, -0.041, -0.04,...	[1.064, 1.057, 1.062, 1.063, 1.054, 1.051, 1.0...	39567
2023-07-11 07:17:45.331897	4	[0.288, 0.288, 0.295, 0.295, 0.292, 0.295, 0.2...	[-0.032, -0.036, -0.036, -0.041, -0.042, -0.04...	[1.062, 1.059, 1.06, 1.056, 1.055, 1.055, 1.05...	39567
2023-07-11 07:17:50.585921	4	[0.289, 0.288, 0.295, 0.295, 0.29, 0.296, 0.29...	[-0.033, -0.037, -0.037, -0.036, -0.04, -0.04, -...	[1.062, 1.055, 1.059, 1.06, 1.055, 1.059, 1.05...	39567
2023-07-11 07:17:56.170992	4	[0.288, 0.292, 0.292, 0.29, 0.288, 0.296, 0.29...	[-0.036, -0.04, -0.036, -0.037, -0.04, -0.036, -...	[1.06, 1.059, 1.055, 1.055, 1.055, 1.056, 1.05...	39567

Figure 10. Data base of 3-axis sensor information

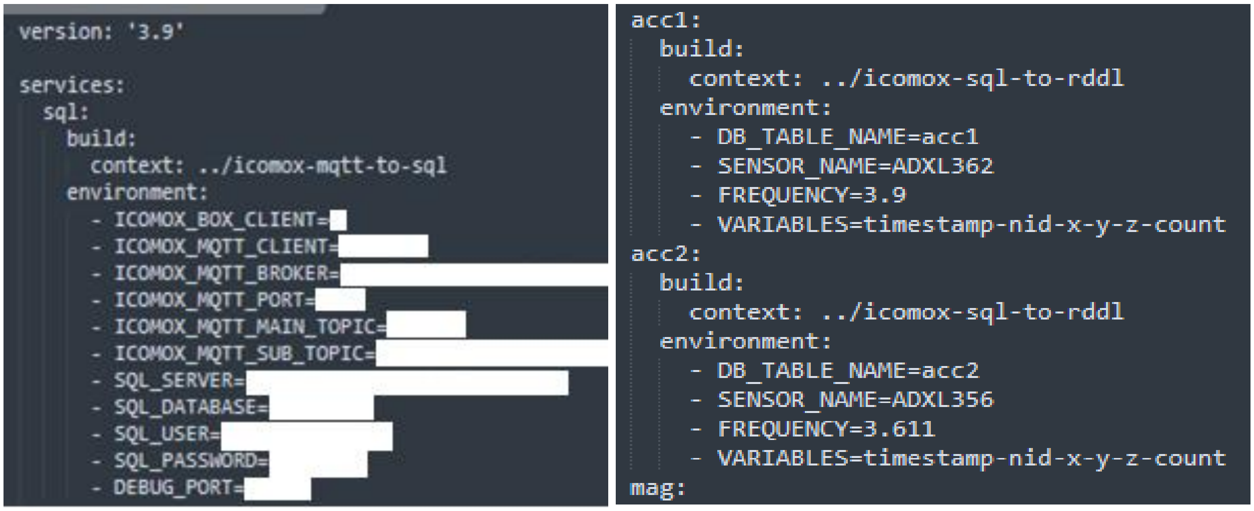
3.3 SQL to RDDL

Once data is stored on the database it is time for the software to run. The code is written on Python language. To ensure continuous working of the software Docker is utilized here.

3.3.1 Docker

Leading containerization platform Docker transforms the creation, deployment, and operation of programs. Since its introduction in 2013, Docker has made it easier to package a program and all of its dependencies into a single, standardized container, guaranteeing consistency and portability in a variety of settings. Docker-created containers include all the code, runtime, libraries, and system tools required for a program to function, which simplifies the deployment of applications across a range of computer environments, from development to production. Docker enables the separation of programs into lightweight containers, facilitating rapid deployment and scalability and a more effective use of resources. Docker has become a vital tool in the domain of DevOps due to its open-source nature and widespread acceptance, allowing developers and operations teams to work more efficiently and improve the software delivery process [24].

For individual sensor separate container is created. Yaml file is used here to create containers. Compose file contains file path where containers should get script, environmental variables. Sensor names demonstrate name of the containers and specific environmental files is added there. The most crucial is variables section. As it is mentioned earlier, the variable name and types differs according to device. They should be dash separated and order should be defined carefully



```
version: '3.9'

services:
  sql:
    build:
      context: ../icomox-mqtt-to-sql
    environment:
      - ICOMOX_BOX_CLIENT=[redacted]
      - ICOMOX_MQTT_CLIENT=[redacted]
      - ICOMOX_MQTT_BROKER=[redacted]
      - ICOMOX_MQTT_PORT=[redacted]
      - ICOMOX_MQTT_MAIN_TOPIC=[redacted]
      - ICOMOX_MQTT_SUB_TOPIC=[redacted]
      - SQL_SERVER=[redacted]
      - SQL_DATABASE=[redacted]
      - SQL_USER=[redacted]
      - SQL_PASSWORD=[redacted]
      - DEBUG_PORT=[redacted]
```

```
acc1:
  build:
    context: ../icomox-sql-to-rddl
  environment:
    - DB_TABLE_NAME=acc1
    - SENSOR_NAME=ADXL362
    - FREQUENCY=3.9
    - VARIABLES=timestamp-nid-x-y-z-count
acc2:
  build:
    context: ../icomox-sql-to-rddl
  environment:
    - DB_TABLE_NAME=acc2
    - SENSOR_NAME=ADXL356
    - FREQUENCY=3.611
    - VARIABLES=timestamp-nid-x-y-z-count
mag:
```

Figure 11. Yaml file structure used to create containers

After containers successfully created, it`s time for software to run. The most important point of script is to define the principles of reading and processing data. In this case it is tested and decided that, the most efficient way of processing is reading it once in a per hour, dividing it into proportions of 10 minutes. After that files for 10 minutes are created and uploaded to data lake. If success message is returned by server, the same amount of data will be deleted from the database. This ensures that size of the database will not be exceeded and we will not get “Could not allocate space”. If there is no data available for next hour, code will go to waiting phase until data comes. This ensures that data communication continues non-stop 24/7, in a most optimized way. Of course as it is in all the project there can be some cases where error arises. This case is also take into account and once docker receives stop signal it triggers specific part of the code which serves as restarting crashed containers.

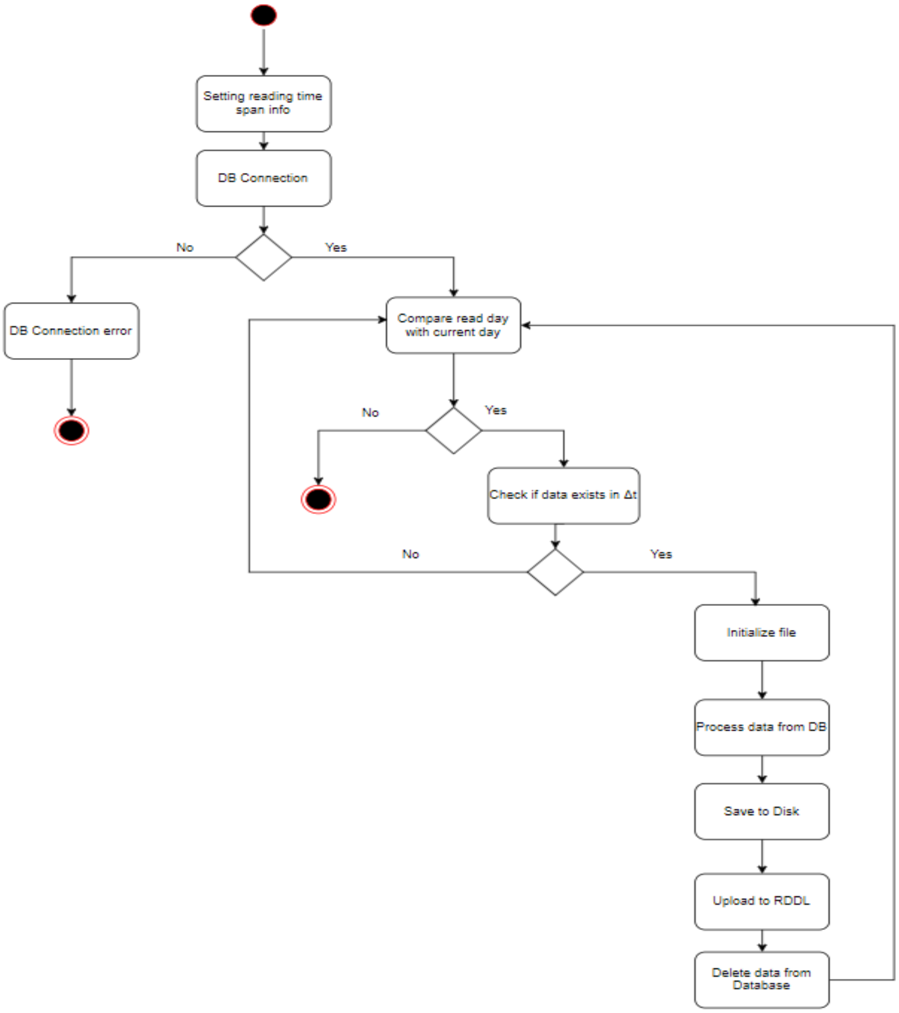
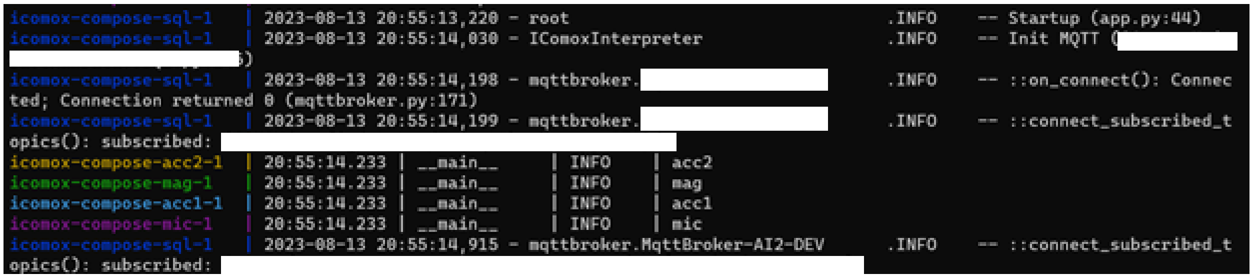
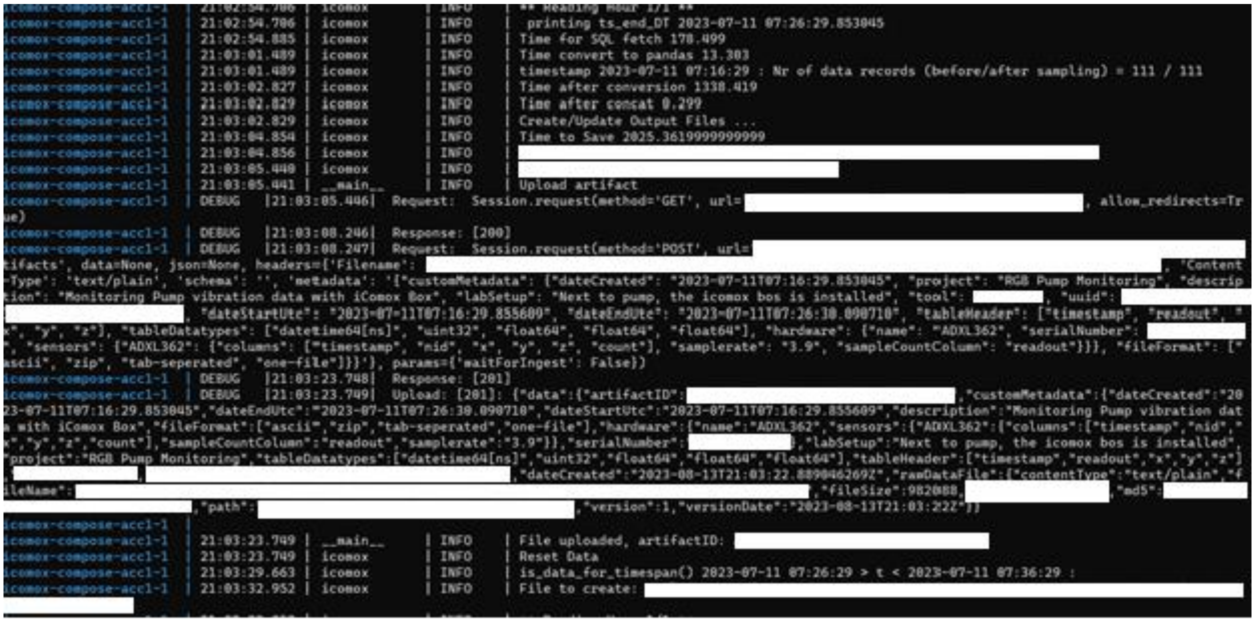

Setting reading time span info
DB Connection
No
Yes
DB Connection error
Compare read day with current day
No
Yes
Check if data exists in Δt
No
Yes
Initialize file
Process data from DB
Save to Disk
Upload to RDDL
Delete data from Database

Figure 12. Graph representation of code data retrieval from MQTT broker



```
icomox-compose-sql-1   | 2023-08-13 20:55:13,220 - root                            .INFO    -- Startup (app.py:44)
icomox-compose-sql-1   | 2023-08-13 20:55:14,030 - IComoxInterpreter               .INFO    -- Init MQTT ([redacted]
)
icomox-compose-sql-1   | 2023-08-13 20:55:14,198 - mqttbroker.[redacted]           .INFO    -- ::on_connect(): Connec
ted; Connection returned 0 (mqttbroker.py:171)
icomox-compose-sql-1   | 2023-08-13 20:55:14,199 - mqttbroker.[redacted]           .INFO    -- ::connect_subscribed_t
opics(): subscribed: [redacted]
icomox-compose-acc2-1  | 20:55:14.233 | __main__      | INFO     | acc2
icomox-compose-mag-1   | 20:55:14.233 | __main__      | INFO     | mag
icomox-compose-acc1-1  | 20:55:14.233 | __main__      | INFO     | acc1
icomox-compose-mic-1   | 20:55:14.233 | __main__      | INFO     | mic
icomox-compose-sql-1   | 2023-08-13 20:55:14,915 - mqttbroker.MqttBroker-AI2-DEV   .INFO    -- ::connect_subscribed_t
opics(): subscribed: [redacted]
```

Figure 13. Logs of Sql connection container creation



```
icomox-compose-acc1-1  | 21:02:54.706 | icomox        | INFO     | ** Reading Hour 1/1 **
icomox-compose-acc1-1  | 21:02:54.706 | icomox        | INFO     |  printing ts_end_DT 2023-07-11 07:26:29.853045
icomox-compose-acc1-1  | 21:02:54.885 | icomox        | INFO     | Time for SQL fetch 178.499
icomox-compose-acc1-1  | 21:03:01.489 | icomox        | INFO     | Time convert to pandas 13.303
icomox-compose-acc1-1  | 21:03:01.489 | icomox        | INFO     | timestamp 2023-07-11 07:16:29 : Nr of data records (before/after sampling) = 111 / 111
icomox-compose-acc1-1  | 21:03:02.827 | icomox        | INFO     | Time after conversion 1338.419
icomox-compose-acc1-1  | 21:03:02.829 | icomox        | INFO     | Time after concat 0.299
icomox-compose-acc1-1  | 21:03:02.829 | icomox        | INFO     | Create/Update Output Files ...
icomox-compose-acc1-1  | 21:03:04.854 | icomox        | INFO     | Time to Save 2025.3619999999999
icomox-compose-acc1-1  | 21:03:04.856 | icomox        | INFO     | [redacted]
icomox-compose-acc1-1  | 21:03:05.440 | icomox        | INFO     | [redacted]
icomox-compose-acc1-1  | 21:03:05.441 | __main__      | INFO     | Upload artifact
icomox-compose-acc1-1  | DEBUG   |21:03:05.446|  Request:  Session.request(method='GET', url=[redacted], allow_redirects=Tr
ue)
icomox-compose-acc1-1  | DEBUG   |21:03:08.246|  Response: [200]
icomox-compose-acc1-1  | DEBUG   |21:03:08.247|  Request:  Session.request(method='POST', url=[redacted]
tifacts', data=None, json=None, headers={'Filename': [redacted], 'Content
-Type': 'text/plain', 'schema': '', 'metadata': '{"customMetadata": {"dateCreated": "2023-07-11T07:16:29.853045", "project": "RGB Pump Monitoring", "descrip
tion": "Monitoring Pump vibration data with iComox Box", "labSetup": "Next to pump, the icomox bos is installed", "tool": [redacted], "uuid": [redacted]
, "dateStartUtc": "2023-07-11T07:16:29.855609", "dateEndUtc": "2023-07-11T07:26:30.090710", "tableHeader": ["timestamp", "readout", "
x", "y", "z"], "tableDatatypes": ["datetime64[ns]", "uint32", "float64", "float64", "float64"], "hardware": {"name": "ADXL362", "serialNumber": [redacted]
", "sensors": {"ADXL362": {"columns": ["timestamp", "nid", "x", "y", "z", "count"], "samplerate": "3.9", "sampleCountColumn": "readout"}}}, "fileFormat": ["
ascii", "zip", "tab-seperated", "one-file"]}}'}, params={'waitForIngest': False})
icomox-compose-acc1-1  | DEBUG   |21:03:23.748|  Response: [201]
icomox-compose-acc1-1  | DEBUG   |21:03:23.749|  Upload: [201]: {"data":{"artifactID":[redacted],"customMetadata":{"dateCreated":"20
23-07-11T07:16:29.853045","dateEndUtc":"2023-07-11T07:26:30.090710","dateStartUtc":"2023-07-11T07:16:29.855609","description":"Monitoring Pump vibration dat
a with iComox Box","fileFormat":["ascii","zip","tab-seperated","one-file"],"hardware":{"name":"ADXL362","sensors":{"ADXL362":{"columns":["timestamp","nid","
x","y","z","count"],"sampleCountColumn":"readout","samplerate":"3.9"}},"serialNumber":[redacted]},"labSetup":"Next to pump, the icomox bos is installed",
"project":"RGB Pump Monitoring","tableDatatypes":["datetime64[ns]","uint32","float64","float64","float64"],"tableHeader":["timestamp","readout","x","y","z"]
,[redacted],[redacted],"dateCreated":"2023-08-13T21:03:22.889046269Z","rawDataFile":{"contentType":"text/plain","f
ileName":[redacted],"fileSize":982088,[redacted],"md5":[redacted]
,"path":[redacted],"version":1,"versionDate":"2023-08-13T21:03:22Z"}}
icomox-compose-acc1-1  |
icomox-compose-acc1-1  | 21:03:23.749 | __main__      | INFO     | File uploaded, artifactID: [redacted]
icomox-compose-acc1-1  | 21:03:23.749 | icomox        | INFO     | Reset Data
icomox-compose-acc1-1  | 21:03:29.663 | icomox        | INFO     | is_data_for_timespan() 2023-07-11 07:26:29 > t < 2023-07-11 07:36:29 :
icomox-compose-acc1-1  | 21:03:32.952 | icomox        | INFO     | File to create: [redacted]
```

Figure 14. Logs of Sensor data processing container

Above pictures contains information, how system works, Docker connection to MQTT broker which it gets data and send it to SQL. Last picture is indication of acc1 sensor container. Each container shows log lines about the what process is finished and how many time did it take to carry it out. Also we can see that there is metadata file and artifactId in the server itself.

So this chain of processes shows us how information is delivered from sensors, transferred using MQTT messaging service, stored in the SQL database, extracted from database, processed and converted into file format, uploaded to data lake and deleted from the database.

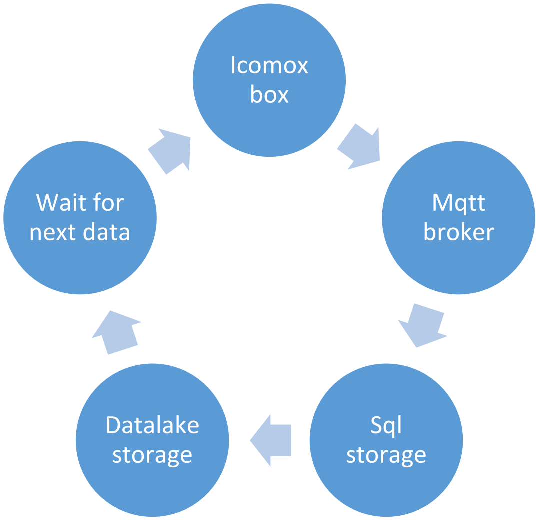


Figure 15. Workflow of data sensing, processing and storing.

4. Analysis and result

After data collected, we are ready to go to implement predictive maintenance. To start it initially we need to have a data set to input ML algorithms. The data collected is not sufficient to work with. To enhance data and have more detailed view of how motors act, signal processing namely indicator extraction should be carried out.

Before starting feature extraction first we need clarify of structure of the data. Depending on the type of sensor data comes 1024, 2048 and 512 portions. After careful analysis I came to conclusion that, combining corresponding number of readings together and doing signal processing on the is the most suitable way.

4.1 Feature extraction

Feature extraction is a critical step in converting raw data into a more manageable and useful representation. Feature extraction is a technique that includes finding and choosing meaningful traits or patterns from input data in domains such as image processing, natural language processing, and signal analysis. These extracted characteristics serve as a condensed and concise representation, capturing vital facts while removing unnecessary aspects. By lowering dimensionality, improving computing efficiency, and focusing on the most discriminative parts of the data, effective feature extraction plays a critical role in boosting the performance of machine learning algorithms. This procedure not only improves knowledge of the underlying structures in the data, but it also helps to more robust and accurate modeling results.

In signal processing and data analysis, time domain feature extraction is an important procedure that includes extracting critical information from signals or time series data in a particular temporal domain. With this approach, one may obtain important insights into the behavior of a signal by measuring its intrinsic properties over time. Statistical metrics including mean, standard deviation, skewness, and kurtosis are common time domain characteristics that provide a thorough knowledge of the central tendency, variability, and shape of the signal. Time domain characteristics can also include zero-crossing rates, signal length, and peak amplitudes, among other metrics that help provide a comprehensive picture of the temporal dynamics. The extraction of these features is critical in a variety of fields, ranging from biomedical signal processing to audio analysis, allowing researchers and engineers to analyze and interpret time-varying data for applications such as pattern recognition, anomaly detection, and classification.

Minimum

A dataset's min function is a mathematical operation that returns the lowest value in the supplied collection of data points. It gives a basic yet critical statistic for determining the bottom bound of the dataset's numerical range, providing insights on the sample's minimal value. This statistical measure is used in a variety of analytical scenarios, such as finding outliers, analyzing the range of results, or creating a baseline for statistical analysis comparison. The min function is widely used in a variety of industries, such as data science, economics, and scientific research, to get essential insights into the lowest observable quantity inside a dataset. Min function helps us to get min of 1024 readings each time. Thus we have information about what min value happened during one reading.

$$\mathrm{Minimum}(X)$$

Maximum

A dataset's max function is a mathematical procedure that calculates the biggest value in a given set of data points. This function is critical for determining the upper limit of the numerical range of the dataset, as it provides vital insights about the greatest value seen within the sample. In statistical investigations, the max function is used to find outliers, examine the range of results, or provide an upper threshold for comparison. The max function is a vital tool for extracting crucial information about the greatest observed quantity inside a dataset in numerous disciplines such as data science, economics, and scientific research, contributing to a full knowledge of the dataset's variability and extremes. Like getting min value, the same operation is done here. Max value discloses information about the maximum from one reading.

$$Maximum(X)$$

Mean

The mean function, often known as the average, is a fundamental concept in mathematics and statistics that is important for summarizing and comprehending data. It is a central tendency measure that provides a single representative value for a group of numbers. The mean is computed by summing all of the values in a dataset and then dividing the total number of values by the total number of values. It is frequently indicated symbolically by the Greek letter (mu) for a population mean for a sample mean. The mean acts as a reference point for the other values in the collection to cluster around. It provides a straightforward and obvious method for determining the "typical" value of a group of data, making it a popular measure in subjects such as economics, physics, and psychology. It is crucial to remember, however, that the mean can be sensitive to extreme values, or outliers, which can have a disproportionate impact on its value. Alternative measures of central tendency, such as the median, may be more appropriate in such instances. Nonetheless, the mean function remains a fundamental tool for summarizing and evaluating data, offering significant insights into a given dataset's central tendency. After summing up all values in one time reading, it is divided by number of elements. Mean value helps us to get overview about overall information, how motor acts.

$$\sqrt{\frac{\sum \mathrm{X}}{N}}$$

Standard deviation

The standard deviation is a statistical metric that offers useful information about the spread or dispersion of a group of data points around their mean. It is an important tool for evaluating the variability within a dataset since it provides a numerical representation of how much individual data points depart from the average. The standard deviation is calculated as the square root of the variance and is represented in the same units as the data, making it easily interpretable. A low standard deviation indicates that the data points are closely grouped around the mean, indicating that the dataset is more consistent and predictable. A higher standard deviation, on the other hand, indicates greater unpredictability, with data points spread throughout a broader range. This statistical measure is frequently used to evaluate and compare the consistency and dependability of data sets in domains like finance, science, and the social sciences. The standard deviation offers a quantitative measure of the degree of dispersion in a dataset, helping researchers and analysts make decisions based on the inherent variability within the data. This is useful when assessing the accuracy of experimental results, analyzing financial risk, or comprehending the distribution of scores in educational assessments.

$$\sqrt{\frac{\sum(x-\mu)^2}{N}}$$

Skewness

Skewness, a basic statistical term, uncovers asymmetry in data distribution, offering critical information about a dataset's structure and tail behavior. This statistic quantifies whether the bulk of observations in a dataset are clustered on one side of the mean more than the other. A skewness of 0 indicates a totally symmetrical distribution, with the left and right sides mirroring each other. Positive skewness indicates that the tail on the right side of the distribution is longer or fatter than the tail on the left, indicating an extended tail towards higher values. Negative skewness, on the other hand, denotes a longer or fatter left tail, implying a stretching of the distribution towards lower values.

Skewness is a flexible technique used in a variety of fields, including biology and economics. For risk assessment in finance, an understanding of the skewness of asset returns is essential. An asset's positive skewness may impact investing strategies and decision-making by suggesting that there are more opportunities for modest profits and fewer for significant losses. Skewness may be used in biology to examine how traits are distributed throughout a population and determine whether some features are more common on one end of the spectrum than the other.

The third standardized moment is used to calculate skewness, which offers a more sophisticated view of the dataset than mean and variance alone. It may be represented as the third standardized instant divided by the standard deviation cube. This formulation accounts for both the size and direction of skewness, providing a complete assessment of the distribution's asymmetry.

However, evaluating skewness necessitates a more complex approach. Outliers, or extreme values, can have a major influence on skewness, which can lead to misunderstanding. As a result, it is critical to combine skewness analysis with other statistical metrics to provide a more complete picture of the dataset's properties.

Skewness is important in statistical analysis because it increases the descriptive power of data. It goes beyond basic central tendency and spread to provide a more detailed view of the form of the distribution. Researchers and analysts obtain a better grasp of the intricacies inherent in real-world data by including skewness into the study. Skewness is a crucial tool in the statistician's toolbox, helping to enhance the interpretation of data distributions and guiding choices across a wide range of fields [25].

$$\sqrt{\frac{\sum(x-\mu)^3}{(\mathrm{N} * \sigma^3)}}$$

Kurtosis

Kurtosis is a statistical metric that investigates the structure and properties of probability distributions in datasets. While mean and standard deviation provide insights into central tendency and dispersion, kurtosis provides a more detailed knowledge of a distribution's tails and extremes. The fourth standardized moment, often known as kurtosis, is an important metric. A mesokurtic distribution with zero kurtosis is similar to a normal distribution in terms of tail and center peak properties. Positive kurtosis, also known as leptokurtosis, displays thicker tails and a more pointed core area, indicating an increased chance of outliers or extreme values. Negative kurtosis, also known as platykurtosis, indicates lighter tails and a larger central peak, indicating a more stable and predictable system.

Kurtosis has implications in a variety of sectors, including decision-making processes and risk assessments. Understanding the kurtosis of a distribution, for example, may assist analyze market volatility and predict extreme occurrences in finance. Kurtosis influences the use of statistical methods and models in data analysis, especially where assumptions regarding normalcy are critical.

Kurtosis may be determined using a variety of approaches, the most common of which being Pearson's and Fisher's formulas. Pearson's kurtosis is based on moments, but Fisher's kurtosis is based on sample moments, which makes it more resilient for lower sample sizes. These approaches are used by researchers and analysts to acquire a full perspective of their data, allowing them to select appropriate statistical techniques and models based on the distributional properties given by kurtosis.

Kurtosis is important in statistical analysis because it provides essential insights into the form of probability distributions. Its use is widespread, impacting decision-making processes, risk assessments, and model selection. Kurtosis improves our comprehension of data distributions by focusing on the tails and extremes, helping researchers and analysts to more educated and context-specific interpretations of their data [25-26].

$$\frac{1}{N}\frac{\sum_{i=0}^{N-1} X_i^4}{rms^4}$$

RMS

The Root Mean Square (RMS) statistic is a fundamental statistical metric that is frequently used across many fields to assess the size or amplitude of a set of data. This measure is very common in physics, engineering, signal processing, and statistics. The RMS value is a method of describing the effective or "root" value of a variable quantity, making it helpful for studying signals whose amplitude varies over time.

The RMS value is determined in its most basic form by calculating the square root of the average of the squared values of a collection of integers. For a given set of values X_1, X_2, ..., X_n, the RMS value (X_{RMS}) is computed using the formula:

$$\sqrt{\frac{1}{N} X_i^2}$$

This formula effectively captures the magnitude of the values while considering both positive and negative components, providing a comprehensive measure of the overall intensity of the signal.

RMS is useful in statistics, where it is used to measure the variability of a set of data, in addition to physics and engineering. In such circumstances, the RMS value acts as a measure of dispersion, allowing for a more complete comprehension of the data distribution.

Similarly, RMS is used in industrial settings to monitor equipment health and detect abnormalities in machinery vibrations. Changes in vibration patterns, as represented in RMS values, might indicate that an item is about to break or malfunction. Anomaly detection algorithms that make use of RMS can offer early warnings, allowing for preventative maintenance and reducing downtime.

RMS is significant because it may express a representative magnitude that accounts for both positive and negative oscillations. The Root Mean Square provides a versatile and broadly applicable metric for describing the effective size of a dataset or signal, whether in studying electrical waveforms, processing signals, or comprehending statistical variability [25].

Peak to peak

Peak-to-peak (P-P) amplitude, a fundamental metric in signal processing, gives useful information about the whole range of oscillations inside a waveform. This measure reflects the difference between a signal's maximum and minimum values, encompassing the complete amplitude span. Peak-to-peak amplitude is widely used in many domains, including electronics, physics, and data processing.

Understanding the peak-to-peak amplitude in electronics is critical for measuring the voltage or current changes in a signal. Engineers use this measure to quantify the entire swing in an alternating current or voltage waveform, which aids in electronic circuit design and analysis. Peak-to-peak amplitude is used in physics to examine oscillations and vibrations because it provides a thorough assessment of the amplitude range within a certain time period.

Anomalies in motors, such as imbalanced loads, misalignments, or worn bearings, can appear as variations in vibration patterns. Monitoring the peak-to-peak amplitude of vibrations helps engineers to discover departures from the norm, allowing them to identify possible problems early. Peak-to-peak amplitude anomaly detection is critical for predictive maintenance techniques because it gives a proactive approach to resolving problems before they develop into costly equipment breakdowns.

A rise in peak-to-peak amplitude may indicate a motor load imbalance or misalignment, whereas a reduction may indicate difficulties such as failing bearings or mechanical wear. These changes are frequently early warning indicators of approaching problems, and continuous monitoring with peak-to-peak amplitude analysis allows for prompt interventions.

When anomalies are detected, such as sudden spikes or erratic changes in peak-to-peak amplitude, the system triggers alerts for further investigation. Its application in monitoring vibrations provides a reliable method for identifying early signs of motor malfunctions, enabling industries to implement preventive measures and maintain optimal operational efficiency. By incorporating peak-to-peak amplitude analysis into motor health monitoring systems, industries can significantly enhance the reliability and longevity of their machinery [25-27].

$$X_{max} - X_{min}$$

Pulse index

In the assessment of rotating machinery health, the output value is derived through the careful consideration of the ratio between the peak value and the mean value of the input signal. This approach involves a thorough examination of the signal's peak height relative to its mean level, offering valuable insights into the characteristics of the signal. Particularly, this method proves highly effective in discerning the frequency of repetitive impulses stemming from bearing defects. Its efficacy becomes most pronounced when dealing with defects that manifest distinct defect frequencies, as observed in cases like a single spall. However, it is essential to note that the method's effectiveness diminishes when faced with scenarios where defect frequencies lack clear distinctions, as often encountered in situations involving multiple defects [28].

$$\frac{\max(|x_i|)}{\frac{1}{N}\sum_{i=0}^{N-1}|x_i|}$$

Waveform factor

In the context of electrical engineering, waveform factor is a critical measure that describes the shape of a waveform. It is defined as the ratio of a waveform's root mean square (RMS) value to its average value. The form factor, which is often used in power systems and electronics, gives useful insights on the departure of a waveform from an idealized sinusoidal shape. This measure is important in anomaly identification, where variations from predicted waveform characteristics might reveal system failures, malfunctions, or anomalies.

The form factor (FF) is mathematically expressed as , where $\frac{rms}{|X|}$ is the RMS vibration and X is the average vibration. For a perfect sinusoidal waveform, the form factor is 22 or approximately 1.414. Deviations from this ideal value can signify distortions or anomalies in the waveform.

Understanding the waveform factor is crucial in various applications, including power systems and electronics, where the quality of the AC waveform is of paramount importance. In power systems, a distorted waveform can lead to inefficiencies and increased heating in electrical equipment.

To determine the quality and stability of electrical power, form factor is widely utilized in power quality monitoring. Power anomalies such as voltage sags, swells, harmonics, or transient disturbances can cause sudden fluctuations or distortions in the waveform, resulting in differences in the form factor. Maintaining the integrity of sent messages is crucial in communication systems. Variations in the form factor of transmitted signals that are unusual might indicate interference, noise, or data corruption. Thus, form factor analysis is used to assure signal transmission dependability.

Machinery in industrial environments frequently creates distinct waveforms. Mechanical difficulties, wear and tear, or approaching breakdowns can all be indicated by anomalies in the form factor of these waveforms. Continuous form factor monitoring assists in predictive maintenance and minimizes downtime.

While waveform factor is an effective tool for detecting anomalies, it is also important to examine the context and features of the individual waveform being evaluated. Some waveforms depart from the ideal sinusoidal shape due to natural changes, and recognizing the predicted variances is critical to preventing false alarms [25].

Margin factor

The margin factor, which is an important element in vibration analysis for rotating machinery, is computed by dividing the peak value by the squared mean value of the absolute amplitude square roots. This trait is critical in the field of equipment health monitoring, presenting different patterns across diverse bearing situations. Notably, the margin factor reaches its optimum for healthy bearings before gradually falling for circumstances such as faulty ball, defective outer race, and defective inner race defects. When examining flaws in a bearing's inner race, the margin factor exhibits the greatest separation ability, making it a significant metric in the identification of probable machinery concerns. This deep knowledge of the margin factor's behavior emphasizes its importance in spotting and distinguishing between different sorts of faults, which contributes to successful predictive maintenance tactics [28].

$$\frac{\max(|x_i|)}{\left(\frac{1}{N}\sum_{i=0}^{N-1}\sqrt{|x_i|}\right)^2}$$

4.2 Synthetic data generation

Electric motors are renowned for their robust and resilient nature, designed to withstand a variety of challenging conditions. Their resistance to external factors, such as temperature fluctuations, vibrations, and varying loads, ensures reliable and consistent performance in diverse industrial applications. However, extracting abnormal data from motors can be a formidable task due to their intricate internal workings. Motor systems often produce vast amounts of complex data, and identifying anomalies within this data requires advanced monitoring and diagnostic tools. The challenge lies in distinguishing between normal operational variations and potentially problematic deviations, as well as interpreting the subtle signals that may precede motor failures. This complexity underscores the importance of employing sophisticated data analysis techniques and sensor technologies to effectively diagnose and address abnormalities, contributing to the overall reliability and longevity of motor-driven systems.

To have ability to train ML algorithms firstly we need to have abnormal data. As we don’t get it from the motors, we need to generate synthetic data. Synthetic data generation is a pivotal aspect of data science, involving the creation of artificial datasets that emulate the statistical characteristics of authentic data. Various techniques are employed to craft synthetic data.

- **Generative Adversarial Networks (GANs):** These models involve a generator network that creates synthetic data and a discriminator network that distinguishes between real and synthetic data. Through iterative training, GANs produce synthetic data that closely resembles authentic datasets.

- **Variational Autoencoders (VAEs):** VAEs are another class of generative models that learn the latent space of real data and generate new samples within that space. This technique allows for the generation of diverse and realistic synthetic data.
- **Data Augmentation:** This involves introducing variations or distortions to existing real data, expanding the dataset by creating slightly altered versions of the original samples. Techniques like rotation, flipping, and scaling are common in image data augmentation.
- **Rule-Based or Parametric Approaches:** Synthetic data can be generated by defining specific rules or parameters governing the features of the dataset. This method allows for more controlled generation, ensuring that the synthetic data adheres to predetermined criteria.

In this project data generated synthetically by using data augmentation technique. By having previous set as a reference certain amount of data added or subtracted correspondingly to create anomalies. Furthermore, if it is going to be directly increasing it becomes considerably easy for ML models to catch outliers. That's why fluctuations added to data intentionally. For motors if signal increases more than 2 standard deviation value it is considered as outlier [10].

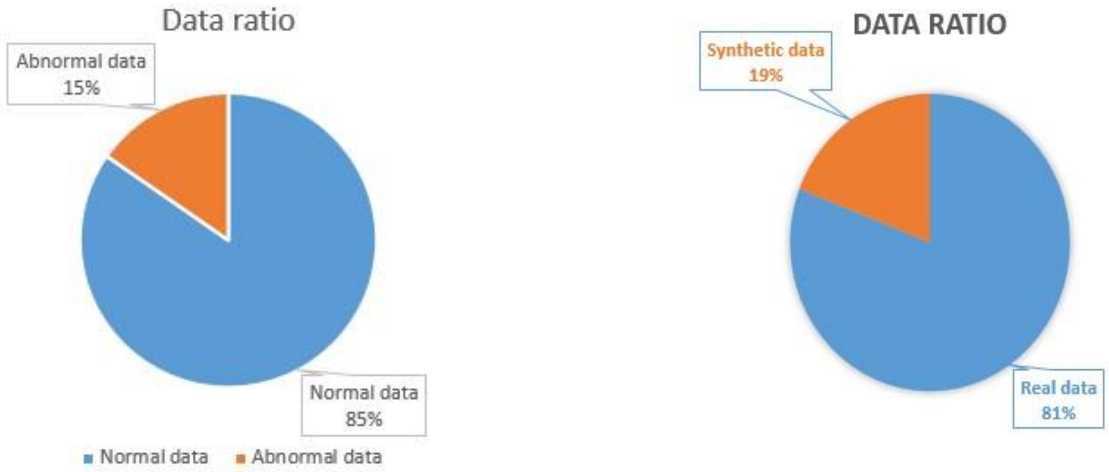


Figure 16. Pie chart indication of proportion of abnormal, normal, synthetic and real data.

Above pie charts demonstrates how synthetic data, real data, abnormal data and normal data are spared proportionally. Roughly 20 percentages of all the data that is used put to ML algorithm is synthetic data. While 4% of it is normal data and 15% is abnormal data. It means that there is also normal data present inside of synthetic data. While all the real data is considered as normal.

The data is not labeled at all, because for as it is mentioned earlier collecting sufficient amount of abnormal data from motors would take more than a half of century. Anomalies usually happens for motor once in a many years. Above picture indicates mean value and number of values dependency. A glance to the graph reveals that until 4400 points data acts as normal. Then data increases until 2std, while it is followed by decreasing to 1 std. Then fluctuation continues until vibration data reaches to break point. Once it is reached vibration value skyrockets and finally pump fails. Overall 5224 data point is used here. 4224 of them are real data, while 1000 of them are synthetic data.

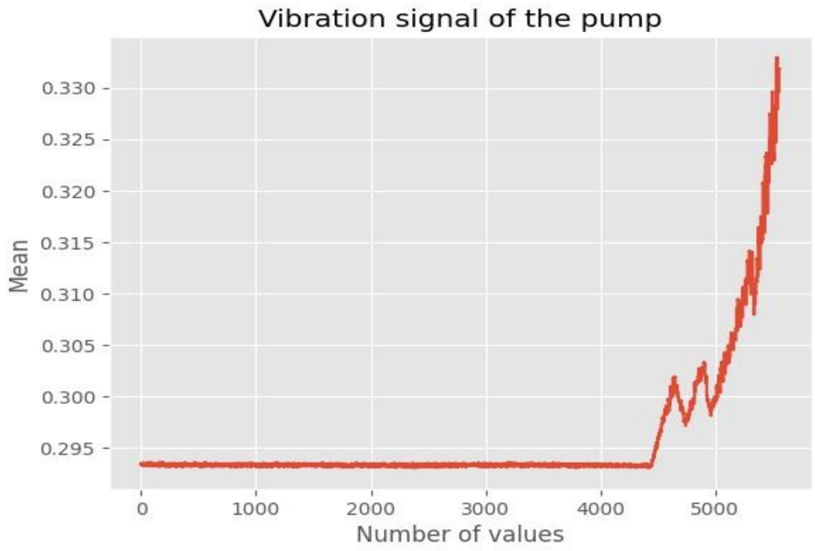


Figure 17. Line graph of input data for developing model

4.3 ML implementation

Machine Learning (ML) is a dynamic and fast expanding technology that enables computers to understand patterns and make predictions or judgments without being explicitly programmed. ML algorithms, at their heart, enable computers to identify complicated patterns in data, adapt to new information, and improve performance over time. This technique is very useful in image and audio recognition, natural language processing, recommendation systems, and autonomous cars.

Supervised Learning: Algorithms are trained on a labeled dataset, where input data is coupled with appropriate output labels, in this sort of machine learning. The model learns to link input features to the proper output, allowing it to anticipate new, previously unknown data.

Unsupervised Learning: In unsupervised learning, computers detect patterns and correlations in unlabeled data without explicit instruction. Clustering, which groups similar data points together, and dimensionality reduction, which simplifies large datasets, are two common activities.

Reinforcement Learning: This kind includes teaching agents to make decisions in a given environment in order to maximize a cumulative reward. Reinforcement learning is frequently employed in circumstances where the model learns by trial and error, making it suited for applications such as robotics, game play, and autonomous systems.

For this application unsupervised learning algorithms is the best choice. Anomaly detection using unsupervised learning algorithms is a powerful approach for identifying irregularities in datasets without the need for labeled instances. Unsupervised methods are particularly valuable in situations where obtaining labeled data for training is challenging or impractical.

K-means clustering algorithm: This unsupervised clustering algorithm combines data points based on similarity, with anomalies recognized as examples that do not correspond to existing groupings.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise): DBSCAN detects anomalies by grouping data points based on density and isolates instances that do not fit well inside high-density zones.

Autoencoders: Autoencoders, a form of neural network, learn data's underlying structure by compressing it into a lower-dimensional representation and then recreating the original input. Higher reconstruction errors are used to find anomalies.

Isolation Forests: This unsupervised technique isolates anomalies using decision trees. Anomalies are cases that need fewer splits in the tree structure to be distinguished from the rest of the data.

SVM (Support Vector Machine): SVM (Support Vector Machine) is a binary classification system that isolates the bulk of data points from a smaller collection of anomalies. It creates a hyperplane that captures usual behavior and classifies cases that fall outside of this boundary as anomalies.

4.3.1 One-Class SVM

The One-Class Support Vector Machine (One-Class SVM) method is a sophisticated machine learning tool created specifically for the job of detecting anomalies in unlabeled datasets. One-Class SVM, which is based on the larger framework of Support Vector Machines (SVM), solves the difficulty of finding cases that vary greatly from the norm, making it especially useful in circumstances where anomalies are uncommon and the bulk of data points indicate typical behavior. This part begins a scientific investigation of the One-Class SVM method, diving into its mathematical formulation, optimization complexities, and major components.

A rigorous mathematical theory at the heart of One-Class SVM tries to locate an ideal hyperplane in a high-dimensional feature space. The main goal is to optimize the distance between this hyperplane and the bulk of data points while allowing for a little amount of variance to accommodate for probable outliers. The optimization problem is expressed as:

Minimize: $\frac{1}{2}\left||\omega|\right|^2 + \frac{1}{\upsilon n}\sum_{i=1}^{n} \xi_i - \rho$

subject to the constraints:

$$\omega * \phi(x_i) \geq \ \rho - \xi_i$$

$$\xi_i \geq 0$$

$$\sum_{i=1}^{n} \xi_i \ = 1$$

Here, ω represents the weight vector perpendicular to the hyperplane, $\phi(x_i)$ denotes the feature mapping of the input data x_i, ρ signifies the offset term, and ξ_i are slack variables that allow for a controlled degree of misclassification. The hyper parameter υ governs the trade-off between the margin width and the number of support vectors, offering users flexibility in tailoring the algorithm to specific detection requirements.

The optimal procedure used in training One-Class SVM is complex, including parts of convex optimization and Lagrange duality. The algorithm attempts to reduce the weight vector's norm while also maximizing the margin and limiting the effect of outliers via the slack variables. The Lagrange multipliers are crucial in this procedure, as they enforce the restrictions and produce the final solution via a dual formulation. This dual problem formulation enables fast computing while also providing useful insights into the algorithm's decision boundaries.

The kernel trick and the outlier score computation are two crucial components that differentiate One-Class SVM. The kernel method allows the algorithm to implicitly transfer input data into a higher-dimensional space, allowing complicated decision boundaries to be accommodated. Various kernel functions, such as the radial basis function (RBF) and polynomial kernels, provide flexibility to a wide range of data properties. Post-training outlier score computation assigns a score to each data point reflecting its divergence from the learnt usual behavior. This score becomes an important statistic for detecting anomalies, with higher scores indicating a greater risk of being an outlier.

One-Class SVM is widely used in anomaly detection across several domains. For example, in fraud detection, when legal transactions far outnumber fraudulent ones, One-Class SVM excels at spotting the rare occurrences of fraud by modeling typical behavior. Similarly, One-Class SVM may detect anomalous patterns that may suggest future breakdowns or malfunctions in industrial systems. Another area where One-Class SVM excels is intrusion detection in cybersecurity, since it can identify aberrant network activity suggestive of possible security breaches.

One-Class SVM`s power rests in its capacity to generalize from the bulk of typical examples seen during training and then recognize anomalies that vary from this acquired normal behavior. This makes it a useful tool in situations where gathering a large dataset of anomalies for training is impossible or where anomalies are naturally sparse. As OCSVM continues to demonstrate its effectiveness in real-world applications, it emphasizes the need of using specialized algorithms for complex tasks such as anomaly detection in machine learning [29].

Nu variable	Signal point	Personal Note	F1 score	AUC-PR
0,01	5563	normal	0,25	0,07
0,05	5552	normal	0,68	0,4
0,1	5518	normal	0,84	0,41
0,2	5473	normal	0,89	0,43

0,5	5412	Gives a lot of false signal	0,92	0,25
0,9	1	Totally wrong	-	-
0,01	5553	normal	0,25	0,07
0,05	5533	normal	0,79	0,4
0,1	5498	normal	0,84	0,44
0,2	5463	normal	0,89	0,43
0,5	5412	Gives a lot of false signal	0,92	0,25
0,9	1	Totally wrong	0,46	0,01

Table 1. Results of different nu variable with 2 different test set.

The data shown here shows the results of a One-Class SVM algorithm over varied nu values, offering light on the complexities of anomaly identification. The first set of runs demonstrates a steady pattern in which nu values of 0.01, 0.05, 0.1, and 0.2 show a progressive increase in model performance. With nu = 0.01, the model looks extremely liberal, yielding a lower F1 score of 0.25 and an AUC-PR score of 0.07. The model gets more discriminating as nu grows, as seen by growing F1 scores (0.68, 0.84, and 0.89) and AUC-PR scores (0.4, 0.41, and 0.43) for nu values of 0.05, 0.1, and 0.2, respectively. These findings imply that the best nu value is somewhere in this range, balancing accuracy and recall.

However, a nu value of 0.5 creates an unusual circumstance. Despite producing a high F1 score of 0.92, it results in a lower AUC-PR score of 0.25. This trend suggests a model that, despite attaining excellent accuracy and recall, suffers from false positives. Nu values of 0.5 and higher produce a more conservative model, resulting in an increased frequency of false signals.

The extreme nu value of 0.9 consistently produces the outcome "Totally wrong." This result underscores the need of avoiding overly high nu values, as they appear to impair the model's ability to forecast accurately.

When the second set of runs is examined, nu values of 0.01, 0.05, 0.1, and 0.2 show a similar trend of incremental increase in model performance. The consistency of these findings across runs emphasizes the One-Class SVM's resilience within this nu range. Notably, the nu value of 0.5 continues to demonstrate a trade-off between a high F1 score and a drop in AUC-PR score, showing that false positives are still a problem. Nu = 0.9 constantly delivers incorrect results, emphasizing its unsuitability for the job at hand.

When the results of both sets of runs are compared, nu values around 0.1 and 0.2 consistently show a balance of precision and recall, making them attractive options for this specific anomaly identification job. The model's sensitivity to nu values emphasizes the need of hyper parameter adjustment and a detailed grasp of the dataset features.

Taking all information into consideration, nu = 0.2 emerges as the best consistent and resilient decision for the One-Class SVM algorithm in this circumstance. Its ability to maintain a balanced approach to accuracy and recall while achieving a high F1 score and a great AUC-PR score across numerous runs proves its dependability in successfully recognizing abnormalities. Further fine-tuning within this range, coupled with ongoing model refinement, may offer additional improvements, providing a foundation for a resilient and effective anomaly detection solution.

Once the nu value is defined next step is to find better performing composition of features. It is defined as combining them and analyzing results.

Indicators	Set name	Nu variable	Signal point	Note	F1	AUC-PR
Mean, Skewness, Kurtosis, Rms, Peak, Variance, Pulse index, Margin factor, Waveform factor	Set 1	0,2	5473	normal	0,89	0,43
Min, Max, Std, Mean, Skewness, Kurtosis, Rms, Peak, Variance, Pulse index, Margin factor, Waveform factor	Set 2	0,2	5463	normal	0,89	0,43
Std, Mean, Skewness, Kurtosis, Rms, Pulse index, Margin factor, Waveform factor	Set 3	0,2	5458	normal	0,9	0,44
Mean, Std, Skewness, Kurtosis, Rms, Pulse index, Margin factor, Waveform factor	Set 4	0,2	5453	normal	0,9	0,44

Table 2. Output of different sets from One-class SVM

Above table contains several runs with a fixed nu value of 0.2 across different sets, each with its own distinct dataset. The examination of these data sets indicates a consistent and reliable performance pattern. The model classified 5473 cases as normal in Set 1, earning an excellent F1 score of 0.89 and an AUC-PR score of 0.43. Set 2 maintained this high level of performance with a comparable nu value, marking 5463 occurrences as normal and repeating the F1 score of 0.89 and the AUC-PR score of 0.43. Set 3 showed a minor improvement, with 5458 normal cases and a slightly better F1 score of 0.9, backed by an AUC-PR score of 0.44. Set 4 identified 5453 occurrences as normal, similar to the previous sets, with an F1 score of 0.9 and an AUC-PR score of 0.44.

The continuous performance with minimal fluctuations across various sets illustrates the stability of the One-Class SVM model with nu = 0.2. The F1 scores, which regularly hover around 0.9, demonstrate a strong balance between accuracy and recall, which is critical for effective anomaly identification. The AUC-PR scores, which measure the model's ability to distinguish between positive and negative occurrences, frequently show respectable results.

While all sets perform well, set 3 stands out significantly more with a slightly higher F1 score of 0.9 and an AUC-PR score of 0.44. This shows that the model's capacity to distinguish anomalies within Set 3's unique features should be improved incrementally.

Finally, while both sets perform well with nu = 0.2, Set 3 has a tiny advantage in terms of F1 and AUC-PR scores, showing a subtle increase in model performance. The selection of the best set, on the other hand, is ultimately determined by the precise priorities and trade-offs relevant to the anomaly detection job at hand. Further investigation and fine-tuning within this nu value range may reveal insights into prospective advancements and optimizations according to the data sets' particular characteristics.

4.3.2 Isolated Forest

The Isolation Forest algorithm provides a paradigm leap in unsupervised anomaly identification, utilizing isolation principles to rapidly find outliers. The Isolation Forest algorithm, developed as an ensemble learning method, illuminates in cases where anomalies are infrequent and different from regular instances. This scientific investigation dives into the algorithm's intellectual roots, implementation complexities, and practical applications.

The Isolation Forest approach is based on the idea that anomalies in a dataset are isolatable points that may be recognized by their ease of separation. The approach generates a set of isolation trees by recursively dividing the feature space. The ease with which anomalies may be isolated enables for their identification with fewer partitioning stages, differentiating them as shorter pathways in tree architectures. The aggregate anomaly score is then calculated based on the average path length across all trees, providing a robust estimate of each data point's oddity.

The Isolation Forest algorithm differs from typical anomaly detection algorithms in that it prioritizes isolation above grouping or density estimates. Isolation trees are constructed by randomly picking a feature and a split value and then building binary divisions until anomalies are isolated. Recursively splitting the data, specifying path length, and aggregating the results over several trees are all part of the mathematical formulation. The mathematical formulation involves defining the average path length E(h) for an isolation tree of height h:

$$E(h) = 2H(h-1) - \frac{2}{n-1}$$

Here, H denotes the harmonic number, and n is the number of instances in the dataset. The anomaly score for a specific instance is calculated as:

$$s(x,n) = 2^{-\frac{E(h(x))}{c}}$$

where h(x) is the path length for the instance x, and c is a normalization factor derived from the average path length of unsuccessful searches in a binary search tree. Instances with lower anomaly scores are considered more anomalous.

Ensemble Learning: The strength of the Isolation Forest lies in its ensemble of isolation trees. By aggregating the anomaly scores from multiple trees, the algorithm enhances robustness and generalizability. This ensemble approach mitigates the impact of outliers and noise, contributing to the algorithm's effectiveness in diverse datasets [30].

Two key components define the Isolation Forest algorithm:

Isolation Trees: The fundamental building blocks, each isolating anomalies through a series of binary splits.

Anomaly Score: The metric quantifying the abnormality of a data point, computed as the average path length across all isolation trees. A lower average path length indicates a higher likelihood of the data point being an anomaly.

iForest has found use in a variety of fields where anomaly detection is crucial. In terms of cybersecurity, iForest can detect strange patterns or behaviors in network traffic, signaling possible security risks or breaches. The algorithm is used in finance to detect fraud by spotting unusual transactions or behaviors that depart from recognized patterns of typical financial activity.

In addition, in industrial systems, iForest can detect abnormalities in sensor data, assisting in the prediction and prevention of equipment failures. Its capacity to operate effectively with high-dimensional data and big datasets makes it a desirable tool for real-world applications requiring the identification of unusual and aberrant occurrences. The Isolation Forest method remains a notable and significant solution as the need for accurate anomaly identification grows across numerous sectors.

iForest's merits are its simplicity, efficiency, and scalability. By integrating the isolation concept with random trees, iForest excels at finding anomalies without the need for enormous computing resources. This section looks at the Isolation Forest algorithm's capabilities, such as its ability to handle high-dimensional data and its processing efficiency. It also covers limits, such as possible sensitivity to the number of trees in the ensemble and difficulties in dealing with skewed datasets.

Indicators	Set names	Signal point	F1	AUC-PRO
Min, Max, Std, Mean, Skewness, Kurtosis, Rms, Peak, Variance, Pulse index, Margin factor, Waveform factor	Set1	5223	0,94	0,79
Max, Mean, Skewness, Kurtosis, Rms, Peak, Variance, Pulse index, Margin factor, Waveform factor	Set2	5182	0,94	0,76
Mean, Skewness, Kurtosis, Rms, Peak, Variance, Pulse index, Margin factor, Waveform factor	set3	5353	0,95	0,8
Mean, Skewness, Kurtosis, Rms, Variance, Pulse index, Margin factor, Waveform factor	set4	5148	0,94	0,74
Mean, Skewness, Kurtosis, Rms pulse index, Margin factor, Waveform factor	set5	5173	0,94	0,76

Table 3. Output from Isolated Forest algorithm for different set of indicators and efficiency scores

Table 3 provides a more detailed look at the performance of the Isolation Forest method over five independent sets, designated set1 through set5. The assessment is based on three main performance metrics: signal point, F1 score, and AUC. Set4 fared particularly well, producing the lowest Signal point at 5148, indicating improved efficiency in isolating abnormalities. Set3, on the other hand, had the highest signal point at 5353, suggesting a lesser effectiveness in anomaly identification.

Investigating the F1 score and AUC metrics offers a more complete picture of algorithmic performance. The F1 scores consistently ranged from 0.94 to 0.95 across all sets, demonstrating a superb balance between accuracy and recall. Similarly, AUC values ranging from 0.74 to 0.80 highlight the algorithm's powerful discriminating capacity. While set4 is the best-performing set in terms of warning point, the algorithm's ability to maintain high standards in anomaly identification is highlighted by the very similar F1 scores and AUC values across all sets.

Understanding dataset-specific performance is crucial for establishing the usefulness of the algorithm. Set4's increased signal point performance demonstrates that dataset properties are linked with the Isolation Forest technique's strengths, possibly showing apparent patterns and unique aberrations. Set3's higher signal point, on the other hand, indicates potential difficulties or complications that might degrade the algorithm's performance. A thorough evaluation of each dataset is necessary to get insight into the factors influencing algorithmic efficiency.

As a result, the Isolation Forest method performs admirably across the datasets presented, with set4 standing out as the best-performing set based on the signal point metric. The algorithm's resilience in anomaly identification is confirmed by the consistently high F1 scores and AUC values across all sets. However, interpreting these indicators requires an awareness of the unique characteristics of each dataset, underlining the importance of continual review and adaptation to enhance performance depending on individual use cases.

4.3.3 DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a sophisticated clustering technique used in data mining and machine learning. Its main advantage is its capacity to find clusters of arbitrary shapes within a dataset while accurately detecting outliers as noise. Clusters are defined by the algorithm as dense regions of data points separated by sparser areas. DBSCAN's capacity to handle datasets with varied densities is one of its primary features, making it particularly helpful in situations where clusters may have unusual forms or sizes.

A significant use of DBSCAN is anomaly detection, which entails extracting strange or unexpected patterns within a dataset, making it vital for jobs like fraud detection, network security, and quality control. DBSCAN, unlike standard clustering algorithms, does not presuppose a fixed number or form of clusters, making it effective at detecting outliers and abnormalities within complicated datasets. The system identifies regular patterns from unusual ones by categorizing points as core, boundary, or noise, resulting in a more detailed comprehension of the data.

The algorithm's mathematical foundation involves two essential parameters: epsilon (ξ), representing the radius within which a data point is considered a neighbor of another point, and

MinPts, the minimum number of points required to form a dense region or cluster. DBSCAN operates by defining three types of points: core points, which have at least MinPts within their ξ-neighborhood; border points, which have fewer than MinPts neighbors but are within the ξ-neighborhood of a core point; and noise points, which are neither core nor border points. Through these definitions, DBSCAN efficiently identifies clusters by traversing the data space based on the connectivity of points [31].

Mathematically, DBSCAN is defined by two crucial parameters: epsilon (ξ) and the minimum number of points (MinPts). The mathematical formulation is concise yet powerful:

Let D be a dataset of n points: $D = \{p_1, p_2, ..., p_n\}$

Core point: p_i is a core point if there are at least MinPts points, including itself, within distance ξ.

Border point: p_i is a border point if it is within distance ξ of a core point but lacks sufficient neighbors to be a core point itself.

Noise point: p_i is a noise point if it is neither a core point nor a border point.

Epsilon	Minimum Samples	Silhouette Score
0.3	3	-0.2355386524652
0.3	5	-0.1477276683125
0.3	7	-0.2118671096103
0.3	10	-0.1995785996113
0.3	20	-0.2201425457351
0.5	3	-0.1181941677774
0.5	5	0.1055118284262
0.5	7	0.02884535451166
0.5	10	0.310335560795
0.5	20	0.1794576055943
0.7	3	0.3198649856949
0.7	5	0.2644157470132
0.7	7	0.2609159579146
0.7	10	0.5818037265544
0.7	20	0.3297726200658
0.9	3	0.3799617857598
0.9	5	0.3800532792282
0.9	7	0.3800207880861
0.9	10	0.595206799947
0.9	20	0.5773397949831

Table 4.Output from DBSCAN algorithm for different Nu variable

The presented dataset summarizes the performance of the DBSCAN clustering method across multiple epsilon (eps) and minimum samples (Minimum Samples) combinations, as measured by the Silhouette Score. This statistic assesses cluster cohesiveness and separation, with higher scores suggesting well-defined and separate clusters.

Among the configurations, the set with epsilon (eps) equal to 0.5 and minimum samples (Minimum Samples) equal to 10 has a Silhouette Score of 0.3103. This suggests that, in this case, a lower neighborhood size correlates to better cluster quality. The epsilon value of 0.5, when paired with an acceptable number of minimum samples, yields clusters with a moderate Silhouette Score, implying a better capture of the intrinsic structure in the data.

Some configurations, on the other hand, have negative Silhouette Scores, indicating poor clustering performance, such as epsilon (eps) equal to 0.3. This indicates that the clusters created by certain parameter combinations are overlapping or poorly defined, and that the algorithm may fail to find significant patterns in the data. It's worth mentioning that in such circumstances, revising the epsilon and minimum sample sizes is critical for improving clustering outcomes.

Furthermore, raising the epsilon value from 0.7 to 0.9 increases the Silhouette Score, suggesting that a larger neighborhood radius might lead to better-defined clusters. However, striking a balance is critical, since overly large epsilon values may result in the creation of a single, all-encompassing cluster.

As a result, the investigation of DBSCAN setups demonstrates a complex link between epsilon, minimal samples, and clustering quality. While the combination of epsilon 0.5 and Minimum Samples 10 performs well, it is critical to examine the unique features of the dataset while establishing the best DBSCAN clustering settings. The negative Silhouette Scores for some setups highlight the importance of fine-tuning parameters to enable meaningful and accurate cluster identification.

Indicators	Set names	Epsilon variable	Number of estimator	Note by me	Silhouette score	Warning point
Min, Max, Std, Mean, Skewness, Kurtosis, Rms, Peak, Variance, Pulse index, Margin factor, Waveform factor	Set 1	0,5	10	normal	0,31	5378

Mean, skewness, kurtosis, Rms, Peak, Variance, Pulse index, Margin factor, Waveform factor	Set 2	0,5	10	normal	0,31	5388
Mean, Skewness, Kurtosis, Rms, Variance, Pulse index, Margin factor, Waveform factor	Set 3	0,5	10	normal	0,31	5403
Mean, Skewness, Kurtosis, Rms, Pulse index, Margin factor, Waveform factor	Set 4	0,5	10	normal	0,31	5417

Table 5. Result from DBSCAN algorithm with efficiency scores

The presented dataset provides insights into the DBSCAN algorithm's performance across four distinct sets, each specified by particular characteristics such as epsilon variable, number of estimators, note, silhouette score, and warning point. The primary goal is to select the set that performs best, with a lower warning point value acting as the major criterion for evaluation.

To begin, when the parameters epsilon variable and number of estimator are examined, it is clear that all four sets have the same values for these parameters, namely 0.5 for epsilon variable and 10 for number of estimator. This consistency implies that the experimental circumstances, such as variable selection and the number of estimators, are constant throughout the sets. This consistency is required for a fair and relevant comparison of the algorithm's performance.

Moving on to the 'Note' column, where all sets are categorized as 'typical,' this shows a shared nature or aim in these trials. However, the label 'normal' is vague, and without more information, it is difficult to determine the possible influence of this variable on the algorithm's performance.

The silhouette score, an important criterion for assessing clustering techniques, is constant across all sets, with a value of 0.31 obtained by each. While this score gives insight into the quality of DBSCAN's clusters, the similar nature of this measure across sets raises concerns about the algorithm's capacity to properly discriminate across clusters. A higher silhouette score normally implies well-defined clusters, and the consistency here suggests that the system may be limited in identifying detailed patterns within the data.

Focusing on the 'Warning point' column, which reflects the notification value, it is stated directly that a lower warning point number equates to a better-performing algorithm. According to this criterion, set 1 is the best-performing of the four, with the lowest Warning point score of 5378.

Sets 2, 3, and 4 are listed in increasing order of related values, further supporting the conclusion that Set 1 surpasses the others based on the provided evaluation criteria.

To conclude, based on the criteria specified, set 1 stands out as the best-performing set among the four in the context of the DBSCAN algorithm, with a smaller warning point value indicating greater performance. To make solid conclusions regarding the effectiveness of each set in a DBSCAN setting, a fuller study would require more information about the algorithm's setup, the dataset, and the experimental aims.

5. Conclusion

5.1 Summary of findings

Thesis focused on predictive maintenance of motors by using the vibration information. The data was collected real time from Air pump that is situated in Regensburg Infineon. Overall 4000 data point processed from 01.04.2023.

Three different unsupervised learning algorithms implemented here. Maim purpose of the research is to detect possible faulty before they happen. To achieve that, 13 features are extracted from the vibration data, which makes up 5213*13 data. Different combinations of the features trained and based on their result, accuracy and efficiency of algorithms measured.

For one-class SVM combination of Std, Mean, Skewness, Kurtosis, Rms, Pulse index, Margin factor, Waveform factor gave the best result. Despite the fact that one-class SVM achieved to detect failure beforehand, the warning message came a little bit late. Additionally, F1 score points out that good result AUC score shows that one-class SVM is not best choice.

Second model performed is Isolated Forest. It is very powerful unsupervised learning algorithm that has ability to find abnormality. Mean, Skewness, Kurtosis, Rms, Variance, Pulse index, Margin factor, Waveform factor is combined together to input the model. They gave the best result in terms of the time. After abnormality signal come at the time where frequency increases 3 standart deviation value. While signal before that can be considered as false signal, and after will be too late to operate. Simultaneously F1 score and AUC score demonstrates good performance.

Third model implemented is another unsupervised learning algorithms DBSCAN. Min, Max, Std, Mean, Skewness, Kurtosis, Rms, Peak, Variance, Pulse index, Margin factor, Waveform factor was brought together and outperformed others. Nevertheless, the time of abnormality signal is not desired.

After all, comparing results of all three models, the best algorithm to be used is Isolated Forest. While the second one is DBSCAN which is followed by one-class SVM. Despite the fact that all 3 algorithm gives some kind of result, efficiency and accuracy of them are still negotiable.

5.2 Limitation of research

Despite different kind of algorithms applied and desired result achieved, there are some limitations for the research.

Firstly, motors are reliable and well performing hardware, which they operate without any problem for years. It is very costly and source consuming task collect anomaly data. The data for different kind of motors and sensors are completely independent, thus it is required to collect data for specific kind of applications.

Secondly, there are number of abnormalities that cannot be foreseen or impossible to conduct during the experiment. So in this case supervised learning algorithms do not work. Sometimes unsupervised approach is lacking in terms of defining the failures.

Thirdly, I executed synthetic data generation method to have abnormal data. Despite it is created according to previous papers and real time data, there can be some problems with simulation. Having real time abnormal data can also improve the model.

Lastly, after reading previous works on this topic two main signal pointed out to have good information about the operation of the system. They are vibration and current. In this project I had very good data for vibration, while current information was not available. By using current information efficiency of algorithm can be improved.

5.3 Recommendations and future research

Initially all previously mentioned limitations need to be taken into account. The data comes only from one motors. Increasing number of motor, can help us to have better understanding of the situation of the motor and improve models accuracy. Furthermore, having some anomaly data for model development can also lead to powerful model.

References

[1] Kerkman R., Leggate D., Pankau J., Schlegel D., Skibinski G., Reflected Waves and Their Associated Current, IEEE Industry Applications Conference, St. Louis, USA, 12-15 Octeber 1998.

[2] O'Donnell P., Report of Large Motor Reliability Survey of Industrial and Commercial Installation, IEEE Transactions on Industry Applications, 1985, IA-21(4), 853-872.

[3] Albrecht B. P. F., Appiarius J. C., Sharma D. K., Assessment of The Reliability of Motors in Utility Applications-Updated, IEEE Transactions on Energy Conversion, 1986, EC-1(1), 39-46.

[4] Jardine, A. K., Lin, D. & Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mechanical Systems and Signal Processing, vol. 20(7), pp. 1483-1510.

[5] P. V. J. Rodrguez and A. Arkkio, "Detection of stator winding fault in induction motor using fuzzy logic," Applied Soft Computing, vol. 8, no. 2, pp. 1112 – 1120, 2008.

[6] Vas, Peter. 1993. Parameter Estimation, Condition Monitoring, and Diagnosis of Electrical Machines. Oxford: Clarendon Press.

[7] Nandi, Subhasis, and Hamid A Toliyat. 1999. "Condition Monitoring and Fault Diagnosis of Electrical Machines -A Review." {IEEE} Ind. Appl. Society Annual Meeting.

[8] Hatzipantelis, E., and J. Penman. 1993. "The Use of Hidden Markov Models for Condition Monitoring Electrical Machines." In Electrical Machines and Drives, 1993. Sixth International Conference on (Conf. Publ. No. 376), 91–96.

[9] M. Amer, M. Goldstein, and S. Abdennadher, "Enhancing one-class support vector machines for unsupervised anomaly detection," in Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description. Association for Computing Machinery, 2013, p. 8–15.

[10] O. A. Egaji, T. Ekwevugbe, and M. Griffiths, "A data mining based approach for electric motor anomaly detection applied on vibration data," 2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), 2020.

[11] R. Ibrahim, R. Zemouri, A. Tahan, F. Lafleur, B. Kedjar, A. Merkhouf, and K. Al-Haddad. Anomaly detection for large hydrogenerators using the variational autoencoder based on vibration signals. In 2022 International Conference on Electrical Machines (ICEM), pages 1609–1615, Sep. 2022. doi:10.1109/ICEM51905.2022.9910728.

[12] Delaunois F., El Kihel B., Jeffali F., Nougaoui A., Monitoring and Diagnostic Misalignment of Asynchronous Machines by Infrared Thermography, Journal of Materials and Environmental Science, 2015, 6(4), 1192-1199.

[13] Mohamed R., Nuawi M. Z., Othman M. S., Vibration and Acoustic Emission Signal Monitoring for Detection of Induction Motor Bearing Fault, International Journal of Engineering Research & Technology, 2015, 4(5), 924-929.

[14] Hulugappa B., Pashab T., Ramakrishnac K. M., Condition Monitoring of Induction Motor Ball Bearing Using Monitoring Techniques, International Journal of Scientific and Research Publications, 2012, 2(11), 406-413

[15] Dennler, N., Haessig, G., Cartiglia, M., and Indiveri, G. (2021). "Online detection of vibration anomalies using balanced spiking neural networks," in 2021 IEEE 3rd International Conference on Artificial Intelligence Circuits and Systems, AICAS 2021 (Washington DC: IEEE), 1–4.

[16] Heistracher, C.; Jalali, A.; Strobl, I.; Suendermann, A.; Meixner, S.; Holly, S.; Schall, D.; Haslhofer, B.; Kemnitz, J. Transfer Learning Strategies for Anomaly Detection in IoT Vibration Data. In Proceedings of the IECON 2021—47th Annual Conference of the IEEE Industrial Electronics Society, IEEE, Toronto, ON, Canada, 13–16 October 2021; pp. 1–6

[17] Analog Devices, Micropower, 3-Axis, ±2 g/±4 g/±8 g Digital Output MEMS Accelerometer, ADXL362, May 2023. https://www.analog.com/media/en/technical-documentation/data-sheets/adxl362.pdf

[18] Analog Devices, Low Noise, Low Drift, Low Power, 3-Axis MEMS Accelerometers, ADXL356, June 2020. https://www.analog.com/media/en/technical-documentation/data-sheets/adxl356-357.pdf

[19] Bosch, BMM150 Geomagnetic Sensor data sheet, BMM150, April 2020. https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bmm150-ds001.pdf

[20] Analog Devices, ±0.5°C Accurate, 16-Bit Digital I2C Temperature Sensor, adt7410, September 2017. https://www.analog.com/media/en/technical-documentation/data-sheets/adt7410.pdf

[21] Infineon Technologies, High performance digital XENSIVTM MEMS microphone, IM69D130, December 2017. https://www.infineon.com/dgdl/Infineon-IM69D130-DataSheet-v01_00-EN.pdf?fileId=5546d462602a9dc801607a0e46511a2e

[22] "MQTT Version 5.0". OASIS. 2019-03-07. Retrieved 2020-12-15.

[23] Beaulieu, Alan (April 2009). Mary E Treseler (ed.). *Learning SQL* (2nd ed.). Sebastopol, CA, USA: O'Reilly. ISBN 978-0-596-52083-0

[24] Ratan, Vivek (February 8, 2017). "Docker: A Favourite in the DevOps World" *Open Source for U*. Retrieved June 14, 2017.

[25] T. W. Rauber, F. de AssisBoldt, and F. M. Varejao, ―Heterogeneous Feature Models and Feature Selection Applied to Bearing Fault Diagnosis, ‖ IEEE Transactions on Industrial Electronics, vol. 62, no. 1, pp. 637–646, Jan. 2015.

[26] J. Chebil, M. Hrairi, and N. Abushikhah, ―Signal analysis of vibration measurements for condition monitoring of bearings, ‖ Australian Journal of Basic and Applied Sciences, vol. 5, no. 1, pp. 70–78, 2011.

[27] M. D. Prieto, G. Cirrincione, A. G. Espinosa, J. A. Ortega, and H. Henao, ―Bearing Fault Detection by a Novel Condition-Monitoring Scheme Based on Statistical-Time Features and Neural Networks, ‖ IEEE Transactions on Industrial Electronics, vol. 60, no. 8, pp. 3398– 3407, Aug. 2013.

[28] Howard, I., 1994, A Review of Rolling Element Bearing Vibration Detection, Diagnosis and Prognosis, Defense Science and Technology Organization, Australia.

[29] Zineb, Noumir; Honeine, Paul; Richard, Cedue (2012). "On simple one-class classification methods". *IEEE International Symposium on Information Theory Proceedings*. IEEE, 2012.

[30] Liu, Fei Tony; Ting, Kai Ming; Zhou, Zhi-Hua (December 2008). "Isolation Forest". *2008 Eighth IEEE International Conference on Data Mining*. pp. 413–422. doi:10.1109/ICDM.2008.17. ISBN 978-0-7695-3502-9 S2CID 6505449.

[31] Schubert, Erich; Sander, Jörg; Ester, Martin; Kriegel, Hans Peter; Xu, Xiaowei (July 2017). "DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN". *ACM Trans. Database Syst*. **42** (3): 19:1–19:21. doi:10.1145/3068335 ISSN 0362-5915 S2CID 5156876