

Master's Programme in Mathematics and Operations Research

# Drift detection methods for data streams

---

Riina Hakkarainen

Master's Thesis  
2023

Copyright © 2023 Riina Hakkarainen

---

<b>Author</b>	Riina Hakkarainen	
<b>Title of thesis</b>	Drift detection methods for data streams	
<b>Programme</b>	Mathematics and operations research	
<b>Major</b>	Systems and operations research	
<b>Thesis supervisor</b>	Prof. Nuutti Hyvönen	
<b>Thesis advisor(s)</b>	Ph.L. Jukka Keisala	
<b>Collaborative partner</b>	Wapice Oy	
<b>Date</b>	<b>Number of pages</b>	<b>Language</b>
29.3.2023	48 + 6	English

---

### Abstract

The main objective of this thesis was to develop an improved anomaly detection method for detecting abrupt and gradual changes in sensor data. The existing method was a user-defined threshold.

Regression-based method was one of the tested methods. Regression lines were fitted separately to the last data points of each sensor, and a sensor was detected as anomalous if its regression line was significantly different from the others.

Bayesian autoencoder was another method that was tested for drift detection. Autoencoder is an artificial neural network that learns a representation of the input data. Autoencoder model was trained using normal sensor data. Drift was detected if reconstruction loss increased suddenly because the reconstruction of the input data did not resemble the original input data.

Statistical tests, such as paired sample t-test and Kullback-Leibler divergence, were also tested for drift detection. Neither of these methods had sufficient performance but including these statistical tests as an additional statistical test in the regression-based method improved robustness of the method.

The best performing method was regression-based method with Kullback-Leibler divergence. It worked well for gradual long-term drift detection which was not as efficient using the existing method. The developed method is robust, but reaction time could be improved.

Two different window sizes were used for detecting fast and gradual drifts. Sliding window size affects reaction times and therefore, adding a method for calculating optimal window size based on the data could improve the performance.

---

**Keywords** drift detection, anomaly detection, regression, Bayesian autoencoder, statistical tests

---

---

**Tekijä** Riina Hakkarainen

---

**Työn nimi** Poikkeamien tunnistaminen aikasarjadatasta

---

**Koulutusohjelma** Matematiikka- ja operaatiotutkimus

---

**Pääaine** Systemi- ja operaatiotutkimus

---

**Vastuupettaja/valvoja** Prof. Nuutti Hyvönen

---

**Työn ohjaaja(t)** FL Jukka Keisala

---

**Yhteistyötaho** Wapice Oy

---

**Päivämäärä** 29.3.2023    **Sivumäärä** 48 + 6    **Kieli** Englanti

---

### Tiivistelmä

Työn tarkoituksena oli kehittää menetelmä poikkeamien tunnistamiseen aikasarjadatasta. Datassa esiintyvät muutokset olivat sekä nopeita että hitaita, joten kehitettävän menetelmän tuli pystyä tunnistamaan erilaisia muutoksia sensoridatasta.

Yksi kehitetyistä menetelmistä oli regressiomenetelmä, jossa regressiosuorat sovitettiin erikseen jokaiselle sensorille. Sensori tunnistettiin poikkeavaksi, jos sen regressiosuora oli merkittävästi erilainen muiden sensorien regressiosuoriin verrattuna.

Toinen kehitetyistä menetelmistä pohjautui Bayesilaiseen autoenkoodajaan. Autoenkoodaaja muuttaa syötteen erilaiseen muotoon ja yrittää rekonstruoida tämän esityksen datasta. Autoenkoodaaja opetettiin käyttäen normaalia sensoridataa. Rekonstruktiovirhe kasvoi, jos autoenkoodajalle syötettiin poikkeavaa dataa, sillä rekonstruktio poikkesi alkuperäisestä syötteestä.

Työssä testattiin myös tilastollisia menetelmiä, kuten riippuvien pariin t-testiä ja Kullback-Leibler divergenssiä poikkeamien tunnistamiseen. Kumpikaan näistä menetelmistä ei toiminut tarpeeksi luotettavasti, mutta nämä menetelmät toimivat hyvin osana regressiomenetelmää.

Paras menetelmä oli Kullback-Leibler divergenssiä hyödyntävä regressiomenetelmä. Kyseinen menetelmä pystyi tunnistamaan hitampia muutoksia tehokkaasti, johon olemassa oleva menetelmä ei soveltunut yhtä hyvin. Menetelmän tarkkuus oli melko hyvä, mutta reaktioaikaa voisi vielä parantaa tulevaisuudessa.

Nopeiden ja hitaampien muutoksien tunnistamiseen käytettiin kahta eri ikkunan kokoa. Automaattinen ikkunan koon määrittäminen voisi kehittää menetelmää, sillä ikkunan koko vaikuttaa merkittävästi reaktioaikaan.

---

**Avainsanat** poikkeamien tunnistaminen, regressio, autoenkoodaaja, tilastolliset testit

---

# Contents

Preface.....	6
1 Introduction .....	7
2 Literature review .....	9
2.1 Anomaly detection .....	9
2.1.1 Data streams.....	9
2.1.2 Internet of Things (IoT) .....	10
2.2 Statistical methods.....	11
2.2.1 Parametric methods .....	12
2.2.2 Non-parametric methods.....	15
2.3 Clustering methods .....	16
2.4 Time series analysis .....	17
2.5 Machine learning.....	17
2.5.1 Supervised learning.....	17
2.5.2 Unsupervised learning .....	18
2.6 Performance metrics .....	20
3 Research material and methods.....	22
3.1 Data preprocessing.....	22
3.2 Regression method.....	23
3.3 Bayesian autoencoder .....	26
3.3.1 Reconstruction loss .....	26
3.3.2 Aleatoric and epistemic uncertainties.....	28
3.4 Statistical tests for change detection .....	30
4 Results .....	32
4.1 Selected methods .....	32
4.2 Simulation testing .....	37
4.2.1 Long-term drift detection.....	38
4.2.2 Short-term drift detection.....	41
4.3 Case study.....	42
4.4 Final method .....	44
5 Summary .....	47
References.....	49

## **Preface**

I want to thank my supervisor Professor Nuutti Hyvönen and advisor Licentiate in Philosophy Jukka Keisala for their guidance. I would also like to thank people I have worked with during this project.

Helsinki, 29 March 2023  
Riina Hakkarainen

# 1 Introduction

The primary research topic of this thesis is to develop an anomaly detection method for detecting a drifting sensor. Drift can be described as an unexpected sudden or gradual change in the sensor measurements. Sensors that are located close to each other are often correlated and therefore, drift can be observed when one sensor behaves differently from other correlated sensors that are measuring the same quantities. Another way of detecting drift is by looking at the past values of the sensor. Drift is detected if recent sensor measurements deviate from the pattern of the past values.

Anomaly detection methods have been researched increasingly over the past few years (Sgueglia et al., 2022, p. 175). Traditional statistical methods have been applied to anomaly detection methods, but machine learning methods have become more common over the recent years based on the literature review. Anomaly detection is important because anomalous sensor values usually indicate that the system is not performing as expected.

The objective of this thesis is to develop an improved drift detection method for a customer that is currently using a user-defined threshold for detecting anomalies. The investigated sensors are measuring physical quantities, such as temperature and pressure, and anomalous sensor readings indicate malfunctions that should be detected. The current monitoring solution for the investigated system is to apply a user-defined threshold for detecting if measurements decrease under a specified limit.

This type of approach has a few challenges. It may be challenging to define the threshold in such a way that anomalies would be detected early without false positives. Threshold-based method is good for detecting sudden drifts but detecting slower drifts may be challenging using a static threshold. However, static threshold is intuitive to the user, and it can be easily adjusted by the user. Objective of this thesis is to test and develop methods that could be used to detect drifts more robustly compared to the current method that is utilized by the customer.

Drift detection method should meet two main requirements, which are robustness and fast reaction time. Anomaly detection method is robust when it correctly detects anomalies without making too many errors. Reaction time should be as fast as possible but reducing reaction time may reduce accuracy of the method because there is less time to verify that detected drift is truly an anomaly. Trade-off between robustness and reaction time is an important topic that should be considered carefully while developing and testing the method.

First, general information about anomaly detection is provided by literature review to give an overview of existing methods. Anomaly detection methods can be divided into few subcategories, such as statistical methods, time series analysis and machine learning methods, which are covered by the literature review.

The most promising methods are selected for a more detailed analysis. Both statistical and machine learning methods are selected for comparison. One topic of analysis is to determine whether an advanced machine learning method outperforms a simpler statistical method. The purpose of this thesis is to also analyse advantages and disadvantages of the selected methods.

The methods will be tested using simulated short-term and long-term drifts as well as few real-world example data sets. Performance metrics and visualizations will be used to compare the methods. The performance of these methods will be tested using number of correctly and incorrectly detected drifts and reaction times. A trade-off between the accuracy of the selected method and reaction time will be considered during the testing to select optimal parameters.

## 2 Literature review

Overview of different types of anomalies is presented in this section. Information about data streams and Internet of things are provided as background information.

### 2.1 Anomaly detection

Anomaly occurs when a single datapoint or multiple datapoints deviate from the expected pattern of the dataset (Foorthuis, 2021, pp. 297-298). These anomalies are often caused by rare and significant events, such as credit card frauds or medical problems (Chatterjee & Ahmed, 2022, p. 4).

Chatterjee & Ahmed (2022, p. 4) describe three main types of anomalies, which are point, contextual and collective anomaly. Point anomaly is a single datapoint that deviates from the rest of the dataset. This type of data point can also be called an outlier (Sgueglia et al., 2022, p. 172). Anomaly is contextual if a datapoint is anomalous only in a specific context. For example, low temperature measurement is normal during winter, but it would be anomalous during summer.

Foorthuis (2021, p. 301) describes that a group of datapoints is a collective anomaly if the group is anomalous when compared to the rest of the data. Qin et al. (2022, p. 1) define that collective anomaly may also occur when multiple sensors are used to measure a system. Sensor values may appear normal when the sensor is observed individually but the measured values can still be anomalous when compared to other similar sensors in the system.

#### 2.1.1 Data streams

Anomalies occur in data streams, which are continuous sequences of datapoints (Erhan et al., 2021, p. 66). Data streams are typically generated by real-time data sources, such as sensors. Anomalies that exist in sensor systems can be classified as spike, noise, constant or drift anomalies. Examples of these different sensor system anomaly types are presented in Figure 1.

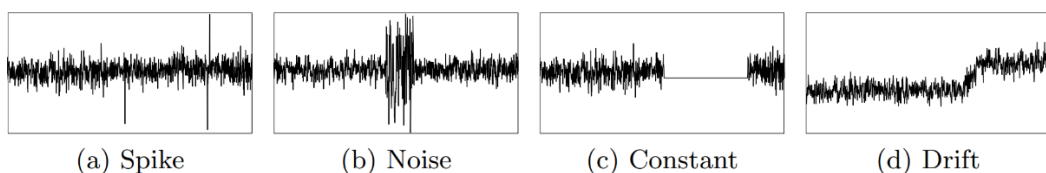


Figure 1: Examples of different sensor anomalies (Bosman, 2016, p. 7).

Spike is a sudden, short duration change in the measured value. Noise occurs when there is unexpected increased variance in the measured values. Constant anomaly happens when the measured attribute stays constant even though the value should change based on the measurement conditions. Drift occurs when offset is observed in the measured values. Drift may occur gradually, and this type of anomaly is usually harder to detect because the change is not sudden (Klein & Verbeke, 2020, p. 392). Gradual drift detection is especially challenging if the anomaly must be detected in near real-time.

Erhan et al. (2021, p. 67) present some common sources of anomalies. One possible source of anomaly is the environment. For example, natural disaster, changing temperature or otherwise unusual condition may affect sensor values gradually or suddenly. These types of anomalies should be detected because they are often caused by important events.

Anomalies may also be caused by errors, such as malfunctioning sensors. Sensors may operate in harsh conditions that may damage the sensors and cause unreliable or incorrect sensor measurements. Sensors can also be exposed to interference, which may cause noise in the measurements.

It is important to separate anomalies caused by error and important events. Anomalies caused by errors are more likely to occur frequently compared to event anomalies (Samara et al., 2022, p. 3). Anomaly detection methods should detect anomalies as well as possible under the assumption that errors and noise are likely to occur in the data. Therefore, anomaly detection methods should be robust so that the number of problems, such as false alarms, can be reduced.

Various malicious attacks, in which external party negatively affects the performance of the system, is another possible source of anomalies (Samara et al., 2022, p. 4). These types of attacks may be hard to detect because the attacked sensor will often behave similarly to a normal sensor. Intrusion detection methods, which are overviewed for example by Khraisat et al. (2019), are beyond the scope of this thesis.

### **2.1.2 Internet of Things (IoT)**

Internet of Things (IoT) refers to devices that use sensors and software to exchange data between other devices or systems over the Internet or other networks (Cook et al., 2020, p. 6481). These devices may be connected over private or public networks. There are many different applications of IoT, and anomaly detection methods are often specific to different use cases. Change point detection methods for streaming data are presented in the following sections.

Change point detection algorithms are typically categorized as online or offline (Aminikhanghahi & Cook, 2017, p. 342). The whole dataset is considered when detecting anomalies using offline methods. Online methods often consider only the most recent data points because it may be challenging to

store all the collected data. Therefore, sliding windows, which include only specific number of the latest datapoints, can be used in IoT applications to reduce storage requirements (Cook et al., 2020, p. 6486).

Cook et al. (2020, p. 6484) describe that data of IoT applications can be categorized into univariate and multivariate data. Univariate data is formed by a sequence of data points that are collected by a single sensor. The collected data often consists of timestamps and scalar values, which represent some measured quantity of the system. Anomaly detection in univariate data is often based on comparing a new observation to the local or global history of the data set. Multivariate data consists of several measured quantities, which are collected using multiple sensors. It is important to also consider the relationship between sensors in addition to the history of the data stream in multivariate data.

Fu et al. (2021, p. 9073) define wireless sensor networks as a collection of connected low power sensors that are not directly connected to the public Internet. There are two different types of monitoring strategies, which are centralized and distributed. In a centralized system, the data is collected from sensors of a wireless sensor network and transferred to the central sink node for anomaly detection. The central node has usually more computational power and therefore, transferring data to the central node enables the use of more complex algorithms.

Distributed anomaly detection strategy means that the sensors have more computational power, and the sensors can communicate with each other (Fu et al., 2021, p. 9073). Therefore, it is possible to detect anomalies using sensor cooperation without forwarding data to the central node. Less data transfer is required to the central node, but available computational power is more limited, which makes it challenging to use more complex algorithms.

Albattah and Rassam (2022, p. 5) define two different types of correlations that should be considered when detecting anomalies in sensor data. Temporal correlation means that the sensor readings are time dependent. Temperature is an example of a scalar quantity that can be temporally correlated for example when recently measured temperatures should be similar. Spatial correlation means that sensor values are correlated with other neighbouring sensor values. Spatial correlation is often used in anomaly detection if correlation between sensor readings can be assumed (Fu et al., 2021, p. 9073). A sensor can be detected as anomalous if its measurements are different from the measurements of other similar sensors.

## **2.2 Statistical methods**

Statistical methods assume that data points are generated based on some statistical model (Erhan et al., 2021, p. 68). Anomaly is detected if data points start to deviate significantly from the expected pattern of the data. This expected pattern of the data can be a distribution, such as a normal

distribution. Test statistics that are used in statistical methods often produce confidence interval, which can be utilized to modify the sensitivity of an anomaly detection method (Chandola et al., 2009, p. 35). Another advantage of statistical methods is that if the distribution estimation is robust, then statistical methods can be used without labeled training data.

Disadvantage of the statistical methods is that they assume a specific distribution because the assumed distribution may not be accurate especially for higher dimensional data (Erhan et al., 2021, p. 68). It may also be difficult to select the right statistical test for anomaly detection. There are two types of statistical methods, which are parametric and non-parametric methods, and these methods are discussed in more detail in the following sections.

### 2.2.1 Parametric methods

Parametric methods are methods in which the underlying distribution is known (Chander & Kumaravelan, 2022, p. 9). The data is generated based on a known distribution, for example a symmetrically distributed normal distribution. An anomaly is detected if a data point significantly disagrees with the data model. Parametric methods can be further classified into Gaussian and non-Gaussian methods.

Gaussian methods assume that data is generated based on a Gaussian distribution and the parameters are estimated using maximum likelihood estimates (Chandola et al., 2009, p. 30). Distance from the estimated mean is considered as the anomaly score, and a data point is detected as an anomaly if the score is larger than a predefined threshold.

It is common to use three standard deviations  $\sigma$  from the mean  $\mu$  as a threshold for determining outliers (Li et al., 2022, p. 2). Three standard deviations account for 99.7% of the datapoints for normal distributions. It is assumed that it is very unlikely that data points are located outside this region. It is also possible to select other value  $n$  besides three so that data point is determined as an outlier if it is located outside the region  $\mu \pm n\sigma$ .

Chandola et al. (2009, p. 30) also present boxplot, which is presented in Figure 2, as a possible metric for detecting outliers using lower quartile  $Q_1$ , median and upper quartile  $Q_3$ . The quantity  $Q_3 - Q_1$  is the interquartile range (IQR), which is used to detect outliers. Any points that are located lower than  $1.5 IQR$  lower than  $Q_1$  and  $1.5 IQR$  higher than  $Q_3$  are considered as outliers.  $Q_1 - 1.5 IQR$  and  $Q_3 + 1.5 IQR$  contains 99.3% of observations and therefore, it is approximately equivalent to using three standard deviations as test statistic for normally distributed data.

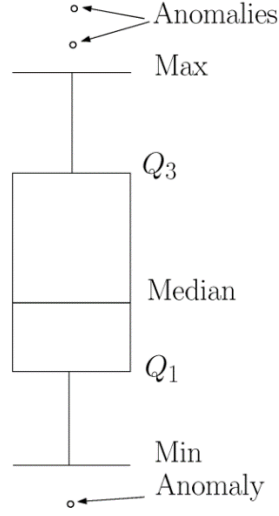


Figure 2: Boxplot (Chandola et al., 2009, p. 31).

Z-score is defined in equation (1) as the distance between data points and the mean divided by the standard deviation (Jamshidi et al., 2022, p. 4). Z-score is positive if  $x$  is above the mean and negative if it is below the mean:

$$z = \frac{x - \mu}{\sigma}, \quad (1)$$

where  $x$  is a single datapoint,  $\mu$  is the mean and  $\sigma$  is the standard deviation.

Pramudita et al. (2019, p. 3) describe modified z-score, which is presented in equation (2), as a possible alternative method for detecting anomalous data points. Modified z-score uses median and median absolute deviation instead of mean and standard deviation:

$$M_i = 0.6745 \frac{x_i - \tilde{x}}{MAD}, \quad (2)$$

where  $x_i$  is a data point,  $\tilde{x}$  is the median and  $MAD$  is median absolute deviation. For a Gaussian distribution,  $MAD$  converges to 0.6745, which is 0.75th quartile of the standard normal distribution. Median absolute deviation is defined using the median (Leys et al., 2013, p. 765):

$$MAD = b * median(|x_i - median(x_i)|), \quad (3)$$

where the constant  $b$  depends on the distribution, and for example for normal distribution it is 1.4826. A datapoint is detected as an outlier if  $|M_i| > D$ , where  $D$  is predefined threshold, such as 3.5.

Zhao et al. (2017, p. 2) argue that the correlation between different sensors in a steady state system should follow statistical distribution, such as normal distribution. Correlations between sensors usually represent the state of a system and a change in the system often affects the correlations between sensors. The Pearson correlation coefficient  $r$ , which is a value between -1 and 1, can be used to calculate correlation between sensors  $x$  and  $y$ :

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}}, \quad (4)$$

where  $x_i$  is a data point of sensor  $x$ ,  $\bar{x}$  is the mean of sensor data  $x$ ,  $y_i$  is a data point of sensor  $y$  and  $\bar{y}$  is the mean of sensor data  $y$ .

These statistical tests are mainly used for univariate time series, but they can also be applied to multiple sensors. Wu et al. (2007, p. 1145) developed an algorithm for detecting outlying sensors. The difference between a sensor measurement and the median measurement of its nearest neighbors is calculated for each sensor. Each calculated difference is standardized before outlier detection. A sensor is classified as an outlier if its absolute value of the standardized difference is larger than a predefined threshold.

Regression model-based methods are also examples of parametric methods (Chandola et al., 2009, p. 32). First, linear regression line is fitted to the data. Anomalies are detected based on the residual, which is the distance between the fitted regression line and a possible anomalous datapoint. Test statistics can be used to determine if the residual is sufficiently large to determine the data point as an anomaly.

Fu et al. (2021, p. 9077) define a fault detection method for correlated sensors in wireless sensor networks. A sensor is determined as normal if its trend is similar to its neighboring sensors and if the median of the measured sensor data is close to the median of its neighboring sensors. Therefore, this method uses a combination of statistical testing and linear regression for anomaly detection. Trend similarity is analyzed using a method, which is based on Pearson correlation coefficient.

Outliers can affect the regression line parameters and therefore, robust regression methods are preferable in regression-based anomaly detection methods. The Theil-Sen estimator, which is named after Theil (1950) and Sen (1968) who published papers on this method, is a robust way of fitting regression line. First, regression slope is calculated between all possible pairs of points and median of these slopes is chosen as the regression slope estimate.

Kajmakovic et al. (2022, p. 13) mention two types of tests for detecting trends in data. Slope-based tests utilize most commonly least squares regression lines. Rank-based tests, such as Mann-Kendall test, are non-parametric tests. Therefore, Mann-Kendall test can be used for any distribution without

having to assume any distribution for the data. It is used for time series in which the trend is constantly increasing or decreasing. Mann-Kendall test is based on calculating differences in signs between later and newer datapoints. If trend is present in the data, the signs will increase or decrease constantly.

Double linear regression is a drift detection method, which is described by Munirathinam (2021, p. 906). Two types of drifts can occur in a data stream, which are short-term and long-term drifts. The algorithm is called double linear regression because there are two separate linear regression models for detecting short-term and long-term drifts.

Drift detection algorithms often use sliding windows because data drift consists of multiple consecutive data points (Klein & Verbeke, 2020, p. 392). Selecting suitable sliding window size is important because drifts should be detected robustly and relatively fast. Increasing window size makes it easier to detect slower drifts that occur during a longer time. However, if the window size is too large, then short-term drift is hard to detect (Munirathinam, 2021, p. 908). Therefore, two different window sizes are useful for simultaneously detecting both short-term and long-term drifts in an accurate manner.

### **2.2.2 Non-parametric methods**

Distribution of the dataset is not assumed to be known in non-parametric methods (Samara et al., 2022, p. 10). Instead, the data distribution is determined from the available data. Anomaly detection is based on distance measures between the datapoint and the statistical model. Examples of non-parametric methods are histogram and kernel functions.

The histogram-based method is a simple non-parametric method for detecting anomalies by keeping track of the normal behavior of the data (Chandola et al. 2009, p. 33). The univariate histogram-based method can be described using two main parts. First, a histogram model is constructed based on the available data. After building the model, new data points are compared with the histogram. If the data point falls into one of the bins, then it is considered as normal. Histogram methods may be prone to false alarms if the bins are too narrow because more data points will be in empty or rare bins. In addition, if the bins are too wide, then anomalous data points may end up in these bins and remain undetected.

Histogram-based methods may be used for multivariate data by constructing histogram separately for each variable (Chandola et al. 2009, p. 34). However, it is difficult to implement histogram-based methods reliably for multivariate data because it is hard to capture interactions between different variables. For example, if attributes appear individually frequently but their combination is rare, histogram-based methods cannot be easily used to detect these types of anomalies.

Samara et al. (2022, p. 10) describe that kernel functions are used to form an approximate probability distribution for the data using normal data. If a new datapoint has low probability density function, then it is considered as an anomaly.

Baldewijns et al. (2016, p. 5) consider cumulative sum as a possible method for detecting shifts in data sets. Cumulative sum is defined as the cumulative sum of differences between data points and target value. Cumulative sum value is positive if the observations are above the target value and negative if observed values are below the target value.

An exponentially weighted moving average can be used for detecting small shifts in the data (Baldewijns et al., 2016, p. 6):

$$z_i = \lambda z_i + (1 - \lambda)z_{(i-1)}, \quad (5)$$

where  $\lambda$  is the chosen weighting factor between zero and one. In this method, the most recent data points have larger weights, which means that these newer data points contribute more to the cumulative sum. Older data points contribute to the weighted sum, but they affect the cumulative sum less because their weights are smaller.

## 2.3 Clustering methods

Clustering methods are used to group similar data points into groups that have similar properties (Samara et al., 2022, p. 11). Clustering methods usually assume that normal data points form a larger cluster, and a deviation from the larger cluster is considered an anomaly. A data point that does not belong to a cluster or few data points that form a significantly smaller cluster can be considered anomalies.

Yu et al. (2020, p. 6) describe an outlier detection method based on K-means clustering. It is an outlier detection method that is based on clustering data points in a sliding window. Outliers are not detected immediately but they are labeled as candidate outliers. The mean value of each cluster is stored, and this metric is compared to calculated metrics in future sliding windows. The candidate outlier must pass several sliding windows before it is determined as an outlier.

Clustering based methods are computationally expensive for multivariate data, and they are difficult to use for detecting continuous changes in data streams (Ayadi et al., 2017, p. 329). A slow drift that occurs in a data stream is an example of a continuous change in a data stream. Therefore, clustering methods are not analyzed in more detail in this work.

## 2.4 Time series analysis

Data streams often generate data as a time series and therefore, predictive methods are also used for anomaly detection (Erhan et al., 2021, p. 68). Anomaly is detected if a new observation is significantly different from the prediction.

Cook et al. (2020, p. 6487) mention different methods that can be used for time series prediction. The autoregressive moving average is one possible method for time series prediction. However, it does not always perform well for non-stationary datasets that have seasonality or non-constant mean.

Zhang et al. (2012, p. 1373) describe an anomaly detection method for wireless sensor networks. This method is based on both spatial and temporal correlations. Each sensor detects outliers based on an autoregressive moving average model. After this detection, the sensor communicates with its neighbors to determine if the observation is also spatial outlier. Zhang et al. (2012, p. 1391) note that especially the spatial calculation is computationally demanding.

## 2.5 Machine learning

Supervised and unsupervised learning are two different machine learning approaches (Erhan et al., 2021, p.69), and these methods are overviewed in the following sections.

### 2.5.1 Supervised learning

Supervised learning is used to train models with known target outputs or labels. These labels are either normal or anomalous. Hilal et al. (2022, p. 5) state that the most common supervised learning method is a predictive model. It is often challenging to get reliable occurrences that represent the anomalous observations. Anomalous occurrences may be rare and therefore, labelled data is hard to gather for training.

An artificial neural network is a supervised data classification method, which consists of many neurons for information processing (Kumar et al., 2019, p. 3). These neurons are inspired by the network of neurons in the human brain. An artificial neural network typically consists of multiple layers that are formed by multiple nodes. The nodes between different layers are connected. There are typically three main types of layers, which are input layer, one or more hidden layers and an output layer. Kumar et al. (2019, p. 3) note that although artificial neural network methods may often have high computational complexity, these methods have been applied in wireless sensor networks for detecting faulty sensors.

## 2.5.2 Unsupervised learning

Unsupervised methods do not use labelled data (Erhan et al. 2021 p. 70). They perform under the assumption that anomalous events are significantly different from the normal occurrences in the data. According to Maleki et al. (2021, p. 2) a common unsupervised method is change point detection which is based on detecting sudden variation in the pattern of a data sequence without using labelled data points.

An autoencoder neural network is an unsupervised learning algorithm, which is trained to reconstruct data as close as possible to the original input data (Jinwon & Sungzoon, 2015, p. 3). Autoencoders have been used for anomaly detection in sensor systems in which anomalies may be caused by real events or broken sensors (Yong et al., 2020, p. 627). Conventional autoencoders have limited ways of quantifying prediction uncertainties, which means uncertainty assessment of these methods is often limited (Yong & Brintrup, 2022, p. 1). Reliable uncertainty estimation is important for ensuring that an anomaly detection method based on an autoencoder is as trustworthy as possible.

An example of autoencoder structure is presented in Figure 3 (Chandra et al., 2022, p. 40484). An autoencoder consists of two parts, which are encoder and decoder (Jinwon & Sungzoon, 2015, p. 3). The encoder maps the original data into a latent representation, and the decoder maps the representation to a reconstructed signal of the input. The difference between the original input  $X = \{x_1, \dots, x_n\}$  and the reconstructed signal  $\hat{X} = \{\hat{x}_1, \dots, \hat{x}_n\}$  is the reconstruction loss.

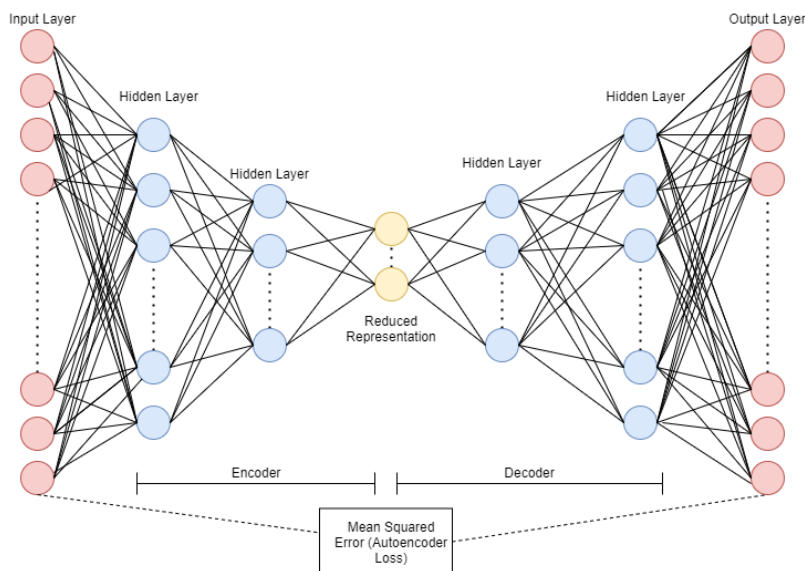


Figure 3: Autoencoder structure (Chandra et al., 2022, p. 40484).

A Bayesian autoencoder is based on Bayes' theorem (Yong et al., 2020, p. 627):

$$P(\theta|X) = \frac{P(X|\theta) P(\theta)}{P(X)}, \quad (6)$$

where  $P(X|\theta)$  is the likelihood,  $P(\theta)$  is the prior distribution of the Bayesian autoencoder parameters,  $P(X)$  is the marginal distribution and  $P(\theta|X)$  is the posterior distribution. The prior  $P(\theta)$  represents the belief about the parameters  $\theta$  before training data  $X$  has been observed (Bernardo & Smith, 2009, p. 2). The likelihood  $P(X|\theta)$  is the probability of obtaining training data  $X$  after observing parameters  $\theta$ . The marginal distribution  $P(X)$  is often assumed to be constant for autoencoders because it does not depend on unknown parameters (Yong et al., 2020, p. 628). The posterior  $P(\theta|X)$  means the probability of observing parameters  $\theta$  based on training data  $X$  (Bernardo & Smith, 2009, p. 43).

Neural networks have usually large number of parameters and therefore, estimating uncertainties using Bayesian frameworks is often difficult in large-scale applications (Pearce et al., 2020, p. 1). Ensembling neural networks is a way of estimating uncertainties by combining estimates of multiple individual neural networks. Neural networks are trained with separate initializations and therefore, some training datasets may be noisier than others. Differences in the training datasets cause variance in the predictions, which can be used to represent uncertainties.

Pearce et al. (2020, p. 2) describe anchored ensembling method, which resembles Bayesian inference method called randomized maximum a posteriori sampling, for approximating the posterior distribution. Model parameters are regularized by drawing values from an anchor distribution. Anchor distribution can be the same as the prior distribution.

An ensemble consists of  $M$  independent autoencoders with a set of parameters  $\theta_m$ , where  $m \in \{1, 2, \dots, M\}$  (Yong et al., 2022, p. 31). Unique weights for each autoencoder are sampled from the prior distribution  $\theta_m^{anc} \sim N(\mu_m^{anc}, (\sigma_m^{anc})^2)$ . The sampled weights remain fixed during training.

The objective is to minimize a loss function consisting of the log-likelihood and log prior during autoencoder training (Yong et al., 2020, p. 628). The log-likelihood and log prior are assumed to be Gaussian. The likelihood loss is:

$$\mathcal{L}(X, \hat{X}) = \frac{1}{N} \sum_{i=1}^N \frac{1}{2\sigma_i^2} \|x_i - \hat{x}_i\|^2 + \frac{1}{2} \log \sigma_i^2, \quad (7)$$

where  $\sigma_i^2$  is the variance of the data point. Typical objective for an autoencoder is to minimize the reconstruction loss  $\|x_i - \hat{x}_i\|^2$ . This corresponds to

a Gaussian distribution with a diagonal covariance matrix in which each variance term  $\sigma_i^2$  is equal to one. The variance term can also be used as a learnable term to estimate the noise level for every data point  $x_i$ . The loss due such a prior is:

$$\mathcal{L}(\theta_j) = \frac{\lambda}{N} \sum_{i=1}^N \|\theta_i - \theta_{anc,i}\|^2, \quad (8)$$

where  $\lambda$  is a hyperparameter, which is used for scaling the regularization term. The total loss function can be formed by combining equations (7) and (8):

$$\mathcal{L}(X, \hat{X}, \theta_j) = \mathcal{L}(X, \hat{X}) + \mathcal{L}(\theta_j). \quad (9)$$

Anchored ensembling is used during training to approximate posterior distribution samples  $\hat{\theta}_m$ , where  $m \in \{1, 2, \dots, M\}$  (Yong et al., 2022, p. 6). These posterior samples are used to calculate  $M$  estimates of the loss function during the prediction phase.

Epistemic uncertainty for a single test data point  $x^*$  can be calculated from the variance of reconstructed signals  $\hat{x}^*$ :

$$Var(\hat{x}^*) = \frac{\sum_{j=1}^M (\hat{x}_j^* - \bar{x})^2}{M}, \quad (10)$$

where  $M$  is the number of ensembled autoencoders and  $\bar{x}$  is the mean of reconstructed signals. Aleatoric uncertainty  $\sigma_i^2$  can be calculated using the log variance of the data, which is returned by the Bayesian autoencoder.

The autoencoder is trained using data without anomalies. After the training, the autoencoder should be able to reconstruct data that resembles the normal data. However, if the data is too different from the training data, the reconstruction error will increase because the autoencoder fails to reconstruct the data back to the input space. The reconstruction error is typically used as an anomaly score in autoencoder-based anomaly detection methods.

## 2.6 Performance metrics

The performance of anomaly detection algorithms can be evaluated by calculating the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) (Bosman et al., 2017, p. 48). True positive is a correctly detected anomaly and true negative occurs when anomaly detection algorithm does not detect an anomaly when there is no anomaly present in

the data. False positive occurs when the algorithm detects anomaly when there is no anomaly in the data. False negative count is the number of times the algorithm did not detect an anomaly when there was an anomaly in the data. Accuracy, precision, recall and F1-score are metrics for evaluating anomaly detection accuracy:

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN}, \quad (11)$$

$$precision = \frac{TP}{TP + FP}, \quad (12)$$

$$recall = \frac{TP}{TP + FN}, \quad (13)$$

$$F1 = \frac{2 * precision * recall}{precision + recall} = \frac{2 * TP}{2 * TP + FP + FN}, \quad (14)$$

where  $TP + FP$  is the total number of detections,  $TP + FN$  is the number of anomalies in the data and  $FP + FN$  is the number of incorrect predictions.

### 3 Research material and methods

Anomaly detection methods for sensor data are presented in this section. The monitored system in the customer's application consists of sensors that are in the same physical location. Therefore, all the sensors are exposed to similar conditions such as outside temperature. Measured sensor values are assumed to be spatially correlated, and some of the anomaly detection methods will be based on this assumption.

The sensors are monitoring different physical quantities at approximately one minute interval. The investigated quantity is formed based on two of these attributes for monitoring purposes. It is assumed that if all the sensors are working normally, rate of change of this quantity should be similar for all sensors. If values of one sensor start to decrease faster compared to the other similar sensors, then it is determined as an outlier and alert is raised.

Data from the sensors is transferred to a cloud where computations occur and therefore, the anomaly detection method for this sensor system is centralized. Hence, there is more computational power available compared to a situation in which computations would have to occur at the sensors.

#### 3.1 Preprocessing

Standardizing the data is one of the main data preprocessing tasks. The mean  $\mu$  of the standardized dataset is zero and its standard deviation  $\sigma$  is one (Braei & Wagner, 2020, p. 25):

$$\hat{x} = \frac{x - \mu}{\sigma}, \quad (15)$$

for all data points  $x$  in the considered set  $D$ . Braei & Wagner (2020, p.25) note that standardization is different from normalization, in which the range of the data is scaled to be between zero and one  $x \in [0, 1], \forall x \in D$ . Normalization is sensitive to outliers and therefore, it should not be used to preprocess data sets that contain outliers. Example data from five sensors after standardization is presented in Figure 4. The measurements of all the sensors decrease at a similar rate which means that the system is performing as expected. Therefore, it is important to consider correlations between the sensors because decreasing values in one sensor is not enough to identify the sensor as anomalous.

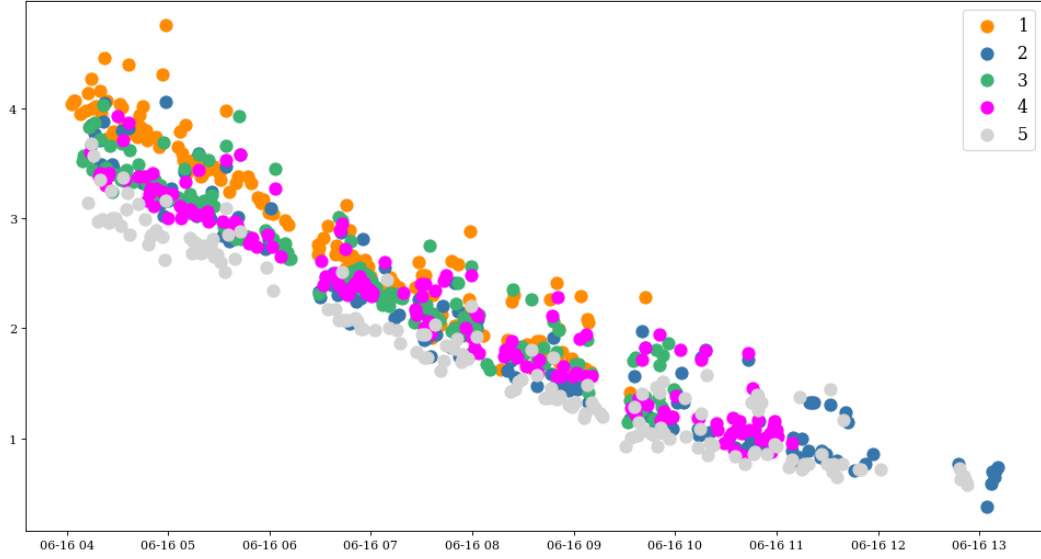


Figure 4: Standardized data from five sensors.

Data streams may contain missing values or outliers which affect anomaly detection. Jamshidi et al. (2022, p. 4) replaced missing values by the mean of three previous values. Maharana et al. (2022, p. 92) describe median and mode as common methods for dealing with missing values. The median is more robust against outliers compared to the mean (Li et al., 2022, p. 4).

It is not necessary to calculate the median of  $n$  previous values in this application based on preliminary testing because calculating the median reduces reaction time for abrupt drift detection, and it is also not necessary for gradual drift detection. However, calculating the median of the  $n$  previous values is one potential method for reducing false positives, and thus, it may be a useful or required addition to the algorithm in the future.

### 3.2 Regression method

Drift in a data set can be analyzed using regression lines, which quantify rate of change in the set. Theil-Sen regression lines are fitted separately to the most recent data points of each sensor. A sensor is anomalous if its regression line is significantly different from the regression lines of other sensors. The hypothesis is that the difference between two regression slopes  $b_i$  and  $b_j$  is approximately zero if the sensors  $i$  and  $j$  are working as expected. Standardized slope difference is used to evaluate the difference between two regression lines (Paternoster et al., 1998, p. 862):

$$Z = \frac{b_i - b_j}{\sqrt{SE_{b_i}^2 + SE_{b_j}^2}}, \quad (16)$$

where  $b_i$  and  $b_j$  are the regression slopes of two sensors and  $SE$  is the standard error of the slope. A standard error for the regression line is calculated based on the confidence interval for the regression slope (Higgins et al., 2022):

$$SE = \frac{\text{upper limit} - \text{lower limit}}{3.92}, \quad (17)$$

where upper and lower limit are the bounds of the confidence interval for the slope, and their difference is divided by 3.92 for the 95% confidence interval. The true regression slope is likely between the lower and upper limit with 95% confidence. If the difference between the upper and lower limit is large, it means that the regression slope estimate is uncertain and thus, the standard error increases. The regression slope estimate is more certain if the difference is smaller and thus, the standard error is also smaller.

Standardized slope differences of a sensor pair seem to approximately follow a normal distribution under normal operating conditions as presented in Figure 5. The normal distribution is centered at zero, which means that the regression lines of these two sensors remain approximately similar during monitoring. Most of the absolute values of standardized slope differences are less than three and therefore, it is selected as the predefined threshold for detecting anomalies.

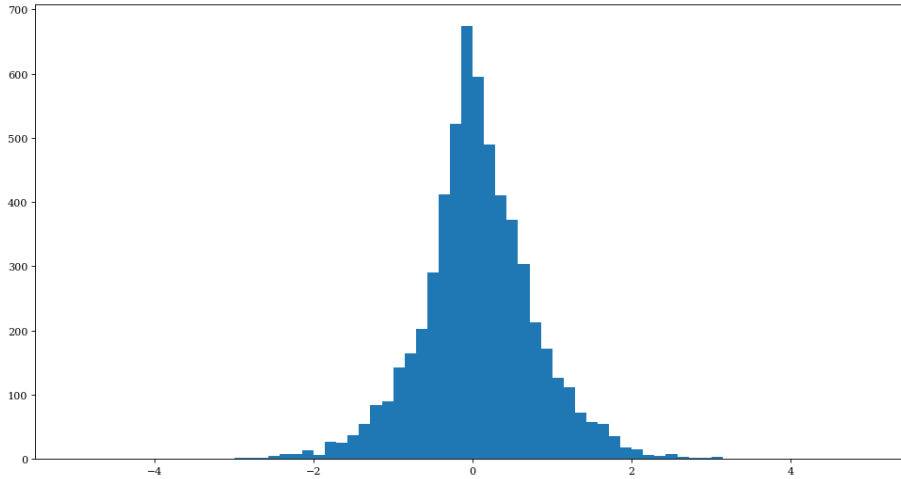


Figure 5: Histogram of standardized slope differences under normal operating conditions.

The main idea of the considered regression method is to detect changes in the distribution of standardized slope differences. The shape of the normal distribution changes if measurements from one sensor start to decrease faster compared to the other sensors. An abnormal decrease in sensor values

affects the shape of the standardized slope difference distribution as presented in Figure 6.

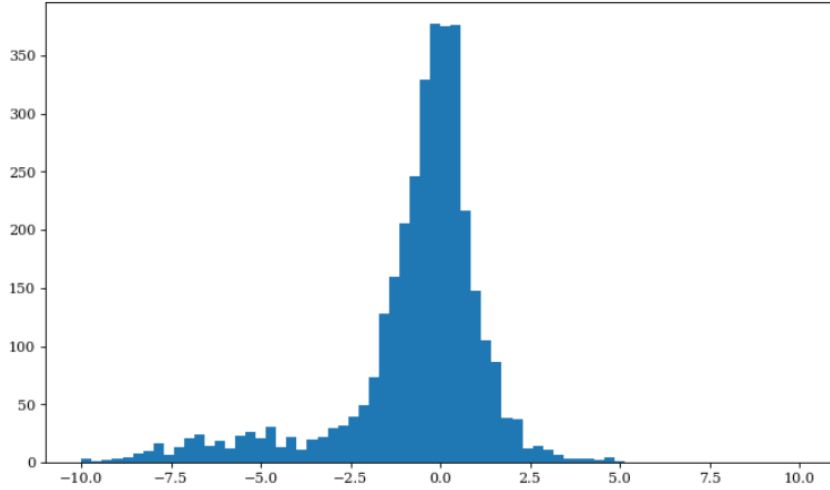


Figure 6: Histogram of anomalous standardized slope differences.

A sensor pair is anomalous if the absolute value of the standardized slope difference exceeds the predefined threshold. The sensor, which has the smaller slope of the pair, is determined as a potential anomaly. Its measurements are decreasing faster compared to the other sensor, which indicates that it is most likely the drifting sensor. A sensor must be significantly different from at least two other sensors so that it can be considered anomalous to avoid false positives. Alert is raised if sensor is detected as an anomaly a few times in a row.

This method can be tested by modifying parameters, such as the size of the sliding window, anomaly detection threshold, and the minimum consecutive anomaly detections. An example of the visualization tool, which is used to evaluate the performance of the method, is presented in Figure 7. The values of the first sensor are modified to decrease faster compared to the other sensors. Alert, which is represented using a red vertical line, is raised after detecting the first sensor as anomalous seven consecutive times.

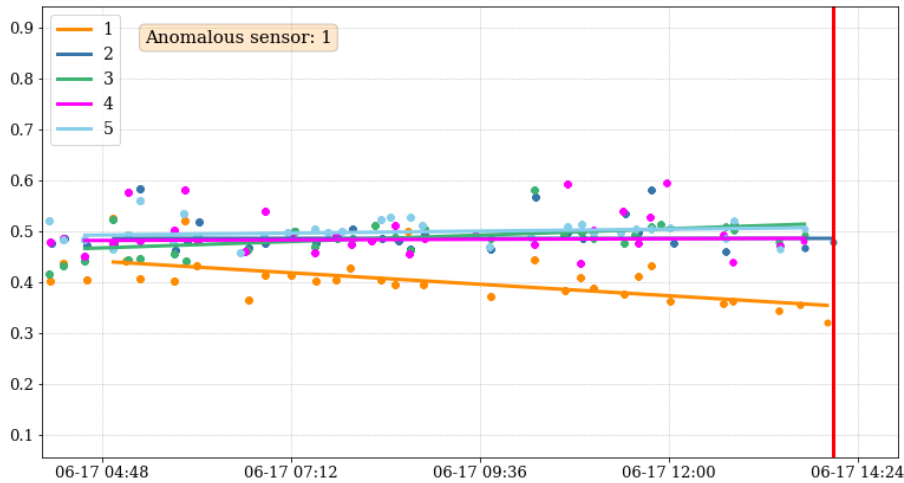


Figure 7: Visualization tool for testing the method.

### 3.3 Bayesian autoencoder

Bayesian autoencoder uses unlabeled training data  $X = \{x_1, x_2, x_3, \dots, x_n\}$  as the input, and the output is a reconstruction of the original signal (Yong et al., 2020, p. 627). The training data consists of predefined number of data points, which are collected during the initialization phase of the algorithm.

A Bayesian autoencoder is trained separately for each sensor using training data in the initialization phase. It is not retrained continuously, and additional initializations may be required because properties of the sensor data change over time. Test data is collected continuously after the initialization phase has ended. It consists of the same number of sensor measurements as the training dataset. The autoencoder reconstructs the test data, and this reconstruction should resemble training data if there is no drift. Sudden increase in the reconstruction loss indicates that training and test data are significantly different.

#### 3.3.1 Reconstruction loss

Reconstruction losses for five sensors are presented in Figure 8. Reconstruction loss increases over time because Bayesian autoencoder is trained only once at the beginning of the monitoring period. Reconstruction loss is quite unstable and there is variation in the reconstruction loss. One possible reason is that the properties of the data change because there are large gaps between measurements. Additional retraining or a larger training set size could produce more reliable results.

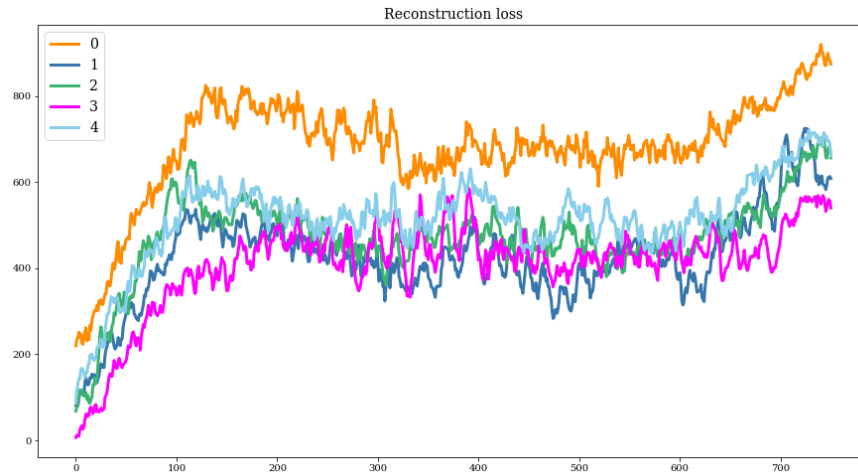


Figure 8: Reconstruction losses during normal operation.

The reconstruction loss should increase for a specific sensor if drift is present in the data. There are a limited number of real test cases and therefore, simulated drifts were used for testing. Reconstruction losses for different drifts are presented in Figure 9. The drifts can be observed as sudden increases in the reconstruction loss after time 500.

Drifts were induced using a multiplier. The smaller the multiplier was, the faster the drift was. For example, if the multiplier was 0.998, then the first sensor value after time 500 was multiplied by this value. The next sensor value after that was multiplied by  $0.998^2$ , and  $n$ th value was multiplied by  $0.998^n$ . Therefore, the multiplier decreased over time causing an induced change in the time series.

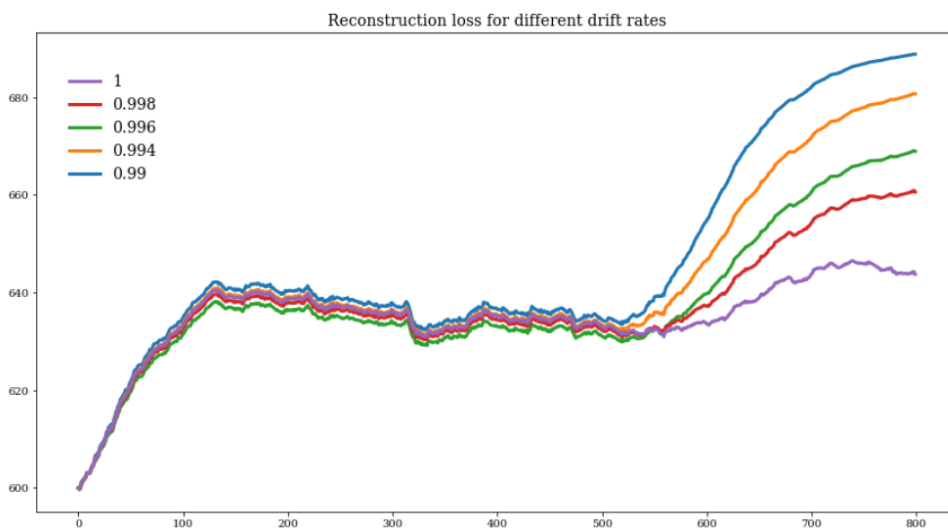


Figure 9: Examples of induced drifts.

A drift is detected if the reconstruction loss starts to deviate significantly from the normal. The reconstruction loss changes over time and therefore, it is hard to define a fixed threshold for detecting anomalous increase in the reconstruction loss. The reconstruction loss also changes between different data sets, and within data sets. Therefore, reconstruction loss should be monitored using sliding windows instead of a fixed threshold.

Selecting optimal window size is important for reliable anomaly detection. One option is to use a fixed window size but choosing optimal window size is often challenging. A small sliding window size works for detecting fast drifts, but a longer window size works better for detecting slower drifts.

Adaptive windowing (ADWIN) is a method in which sliding window size varies based on the rate of change in the data (Bifet & Gavaldà, 2007, p. 443). The size of the sliding window is increased if no change has been detected in the data. In contrast, the size of the sliding window is decreased when a change is detected. The current window is split into two parts and the means of these two windows are calculated. If the difference between these two means is larger than a predefined threshold, change in the data is detected.

Adaptive sliding window method was tested for detecting changes in reconstruction loss. There were many false positives and therefore, a fixed-size sliding windows will be used for detecting changes in the reconstruction loss.

### **3.3.2 Aleatoric and epistemic uncertainties**

Two types of uncertainties, epistemic and aleatoric, are of interest when evaluating quality of predictions (Yong & Brintrup, 2022, p. 2). Epistemic uncertainty refers to uncertainty of the model parameters due to limited training data. Aleatoric uncertainty refers to the randomness of the data, which cannot be explained using models. Aleatoric uncertainty for five sensors is presented in Figure 10. Aleatoric uncertainty stays mostly constant and there is little variation.

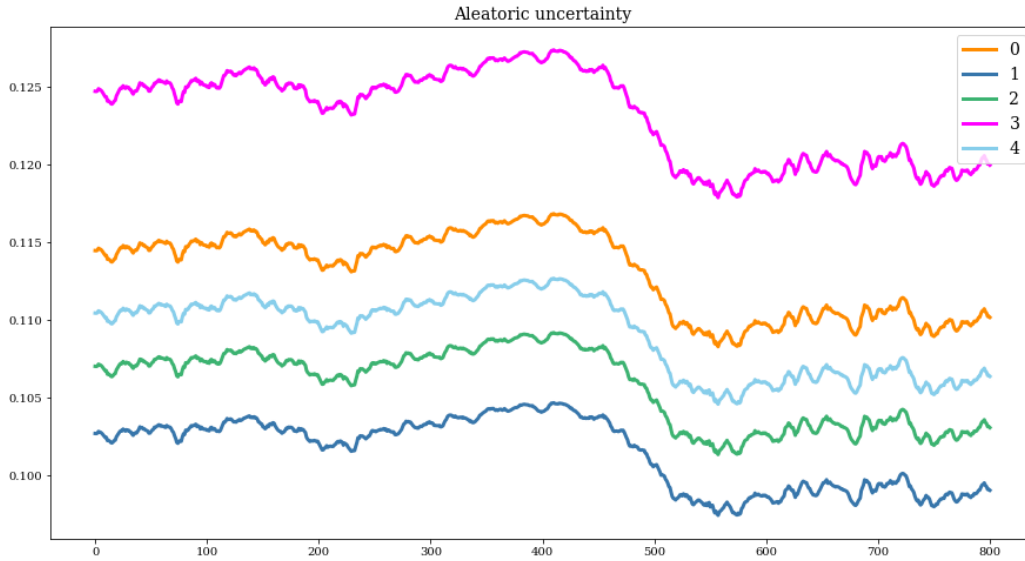


Figure 10: Aleatoric uncertainty.

Epistemic uncertainties, which were calculated using training size of 100, are presented in Figure 11. Epistemic uncertainty fluctuates more compared to the aleatoric uncertainty. Training size of 100 is quite small, which partly explains the amount of noise in the epistemic uncertainty.

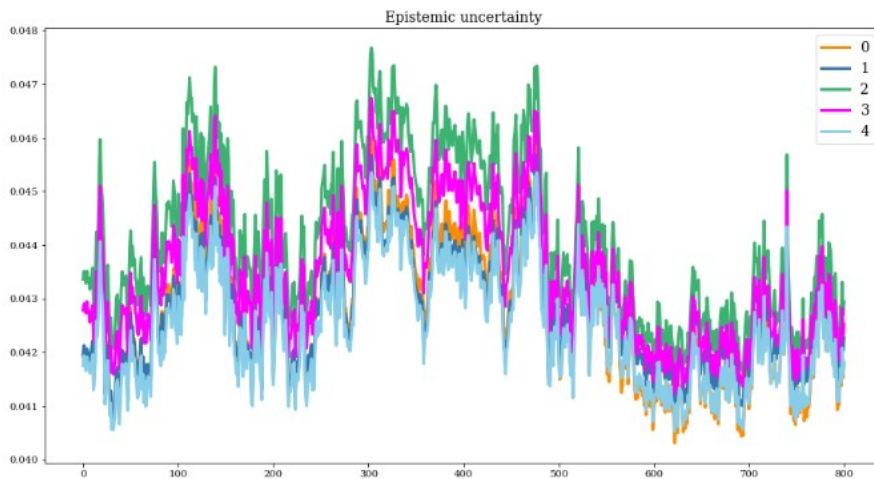


Figure 11: Calculated epistemic uncertainties using training size 100.

Epistemic uncertainties, which were calculated using a larger training size of 400, are presented in Figure 12. Increasing the training data size to 400 considerably decreased the variation in the epistemic uncertainty. Hence, epistemic uncertainty behaves as expected since it describes uncertainty due to limited training data.

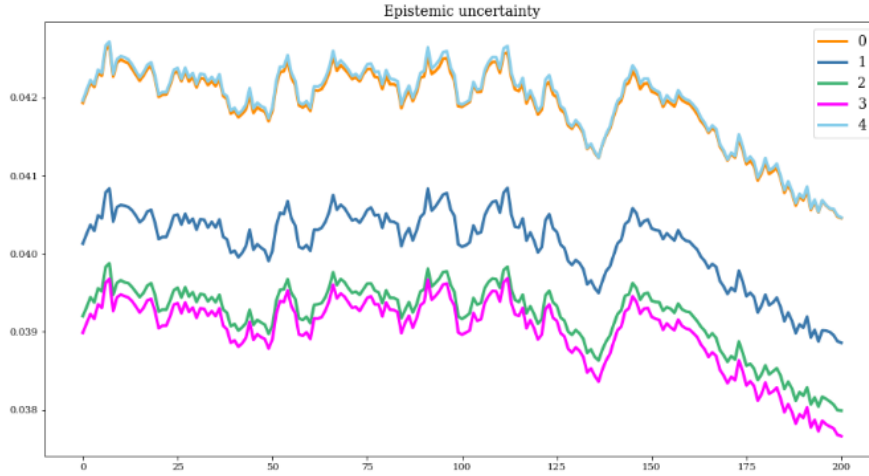


Figure 12: Calculated epistemic uncertainties using training size 400.

Total uncertainty, which is formed by calculating the sum of aleatoric and epistemic uncertainties, can be used for rejecting uncertain anomalies. Uncertainty estimation is important because it enables rejecting uncertain anomalies, which reduces the number of false positives.

### 3.4 Statistical tests for change detection

It is also possible to detect changes in time series by comparing properties of the data using reference and detection windows. A detection window includes the most recent data, and a reference window corresponds to older data. If drift is present in the data, properties of the detection window, for example mean or the probability distribution, should be different from the reference window.

The paired sample t-test can be used to analyze whether there is difference in the means of two dependent samples (Rietvel & van Hout, 2017, p. 46). For example, effectiveness of a treatment can be evaluated using paired sample t-test by measuring participants at two points in time. The idea of using paired sample t-test at different times can be applied to sensor data.

The two-sample Kolmogorov-Smirnov test can be used to compare underlying distributions of two samples by estimating if the data samples come from the same population distribution (Porwik & Dadzie, 2022, p. 170):

$$D_{n,m} = \sup_t |F_n(t) - F_m(t)|, \quad (18)$$

where  $F_n$  and  $F_m$  are the empirical distribution functions of the samples. The Kolmogorov-Smirnov test uses a significance level to determine whether difference between two samples is significant. If p-value is less than 0.05, then

the difference can be determined as statistically significant. If p-value is over 0.05, then the original hypothesis that the differences between two samples are not significant cannot be rejected.

The Wasserstein distance (Ramdas et al., 2017, p. 4) is a distance function, which can be used to calculate a distance between two probability measures:

$$W_p(P, Q) = \left( \inf_{\pi \in \Gamma(P, Q)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|X - Y\|^p d\pi \right)^{\frac{1}{p}}, \quad (19)$$

where  $\Gamma(P, Q)$  is the set of probability distributions on  $\mathbb{R}^d \times \mathbb{R}^d$  whose marginals are  $P$  and  $Q$  on the first and second factors. The first Wasserstein distance is used in this application, which means that  $p = 1$ . The Wasserstein distance can also be described as the minimum cost required to turn one probability distribution into another. Muskulus & Verduyn-Lunel (2011, p. 47) argue that general linear programming solvers can be used to solve optimal transportation problems of weighted point sets after the measures have been discretized.

The Kullback-Leibler divergence is a distance measure that can be used to estimate differences between two probability distributions (Johnson & Sinanović, 2001, p. 1):

$$D(p_1 \| p_0) = \int p_1(x) \log \frac{p_1(x)}{p_0(x)} dx, \quad (20)$$

where  $p_0$  and  $p_1$  are two probability distributions. Note that  $D(p_1 \| p_0) \geq 0$ , with  $D(p_1 \| p_0) = 0$  if and only if  $p_1 = p_0$ . The Kullback-Leibler divergence is not symmetric, i.e.,  $D(p_1 \| p_0) \neq D(p_0 \| p_1)$  in general (Johnson & Sinanović, 2001, p. 2).

## 4 Results

Comparison of the main methods, which are the regression method and the selected statistical tests described in section 3.4, are presented in this section. Performance metrics, such as precision and recall, as well as reaction times are used to compare these methods. The best statistical tests are selected for further testing, and results of these further tests are presented in section 4.2.

Drift detection methods were mainly tested using data sets that did not contain drifts. Therefore, drifts were artificially induced in these data sets for testing purposes. Induced drifts made it possible to test these studied methods using variety of different drifts. A case study was also performed using data sets that contained real-life examples of drifting sensors.

### 4.1 Selected methods

The regression method was evaluated as one potential method based on the preliminary tests conducted in section 3.2. Regression lines quantified drifts in sensor data quite well and adjustable parameters made it possible to change the reaction time and accuracy of the method.

Bayesian autoencoder was not considered further based on initial testing, which was carried out in section 3.3. The reconstruction loss reacted quite well to induced drifts in the data sets, but it was also quite unstable metric in many examples. Bayesian autoencoder was also evaluated as a computationally expensive method and therefore, only statistical methods were tested further.

The methods were tested using both short-term and long-term simulated drifts by inducing different drifts to one of the sensors. Drifts were induced using multipliers 0.95, 0.97, 0.99, 0.996, 0.997 and 0.998 starting at time 700 for all data sets. After time 700, the first value of sensor 1 was multiplied by a multiplier, such as 0.998. The second value after that was multiplied by  $0.998^2$  and the  $n$ th value after that was multiplied by  $0.998^n$ . Therefore, the multiplier decreased over time causing an induced drift in the time series.

An example of a normal data set that does not include a drifting sensor is presented in Figure 13. An example of induced abrupt drift is presented in Figure 14. Blue vertical line indicates time when the short-term version of the algorithm detected the drift. An example of gradual drift is presented in Figure 15. Dashed blue line indicates time when the drift was detected using the long-term version of the algorithm. The other multipliers 0.97, 0.99, 0.996 and 0.997 resulted in drifts that varied between the most gradual drift and the most abrupt drift.

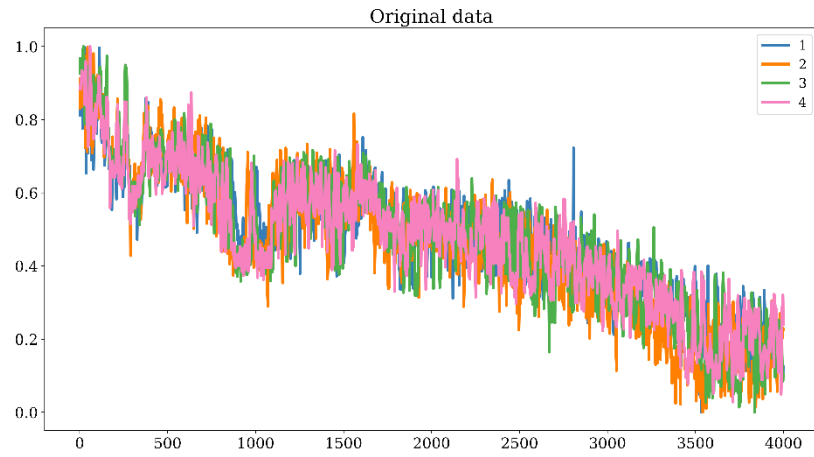


Figure 13. Original data before drift is induced.

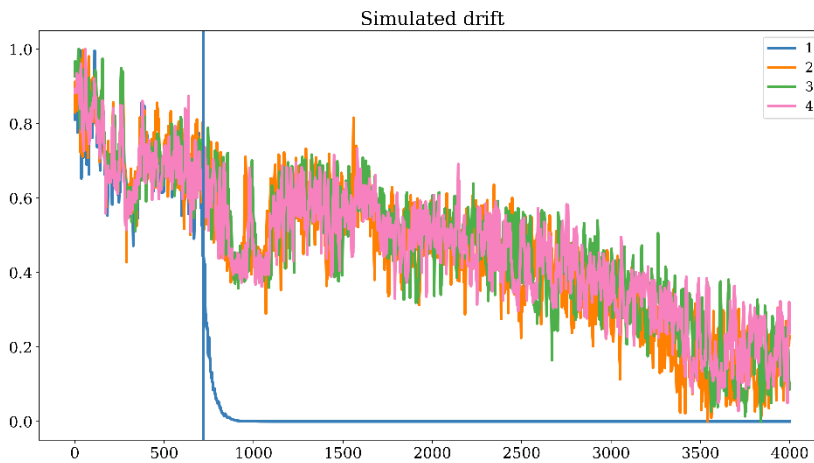


Figure 14. Data after inducing abrupt drift using multiplier 0.95.

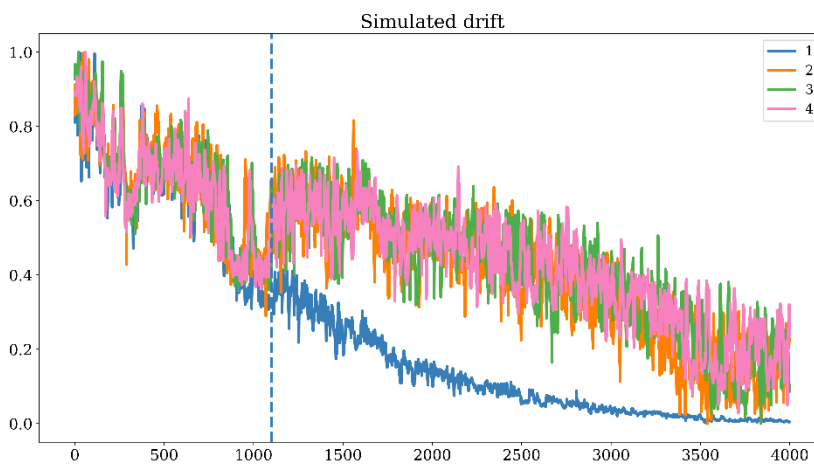


Figure 15. Data after inducing gradual drift using multiplier 0.998.

The selected statistical tests were described in section 3.4, and these test statistics were calculated using reference and recent windows. The reference and recent windows were both half the size of the sliding window, which was used to fit the regression lines. The reference and recent windows were also normalized so that they would resemble probability distributions because some of the selected statistical tests are meant to estimate differences between two probability distributions.

The paired sample t-test was calculated using Python library SciPy, which is a scientific computation library (Virtanen et al., 2020, p. 261). The paired sample t-test was calculated using function called `ttest_rel`, which can be found in the sub-package `scipy.stats`. The Kolmogorov-Smirnov test and the Wasserstein distance were calculated using `scipy.stats` sub-packages called `kstest` and `wasserstein_distance`, respectively. This version of the Wasserstein distance is the first Wasserstein distance, which means that  $p = 1$  in equation (19). These statistical tests take the reference and recent windows as the input and return a test statistic.

The Kullback-Leibler divergence score for this data was calculated according to equation (20), where  $p_0$  is the recent window and  $p_1$  is the reference window. Because division by zero is undefined,  $\varepsilon = 0.00001$  was added to both  $p_0(x)$  and  $p_1(x)$  to avoid dividing by zero in this application.

The tests were evaluated using accuracy, precision, recall and F1-scores. The results are presented in Table 1. These metrics were calculated based on 150 test cases, in which one sensor was contaminated with different drifts and other sensors were unmodified. Therefore, most of the examples did not contain drifts. The purpose of this test was to evaluate if the selected methods detect a change in the right sensor at the right time.

Table 1: Accuracy, precision, recall and F1-score for different statistical tests.

	Accuracy	Precision	Recall	F1-score
The paired sample t-test	91.0	93.9	86.1	89.8
The Kolmogorov-Smirnov test	76.0	51.2	84	63.6
The Wasserstein distance	91.7	77.3	94.4	85.0
The Kullback-Leibler divergence	93.8	93.5	80.6	86.6

The Kolmogorov-Smirnov test had more false positives compared to the other methods. It also had the lowest precision, which means that fewer detected positives were actually positive compared to the other methods. The

Kolmogorov-Smirnov statistic was not able to clearly differentiate between anomalous and normal sensors as presented in Figure 16 and therefore, this statistic was prone to false positives.

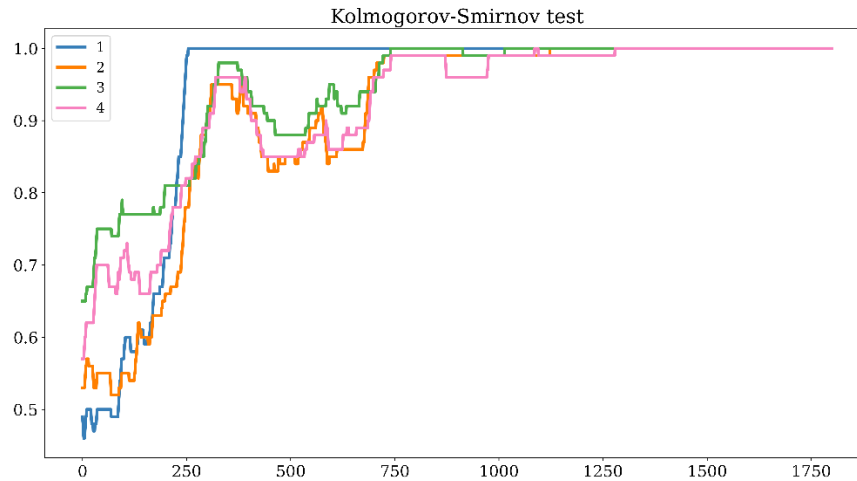


Figure 16. The Kolmogorov-Smirnov test statistic for the data presented in Figure 15.

The Wasserstein distance worked quite well, but it had lower precision compared to the paired sample t-test and the Kullback-Leibler divergence. A fixed threshold was used when detecting changes using the Wasserstein distance. The performance of this distance metric could presumably be improved by using sliding window analysis. For example, a drift could be detected if Wasserstein distance metric is higher than three standard deviations compared to previous reference value. An example of the Wasserstein distance for the data presented in Figure 15 is presented in Figure 17. An unexpected increase in the values of the test statistic can be clearly observed for the anomalous sensor.

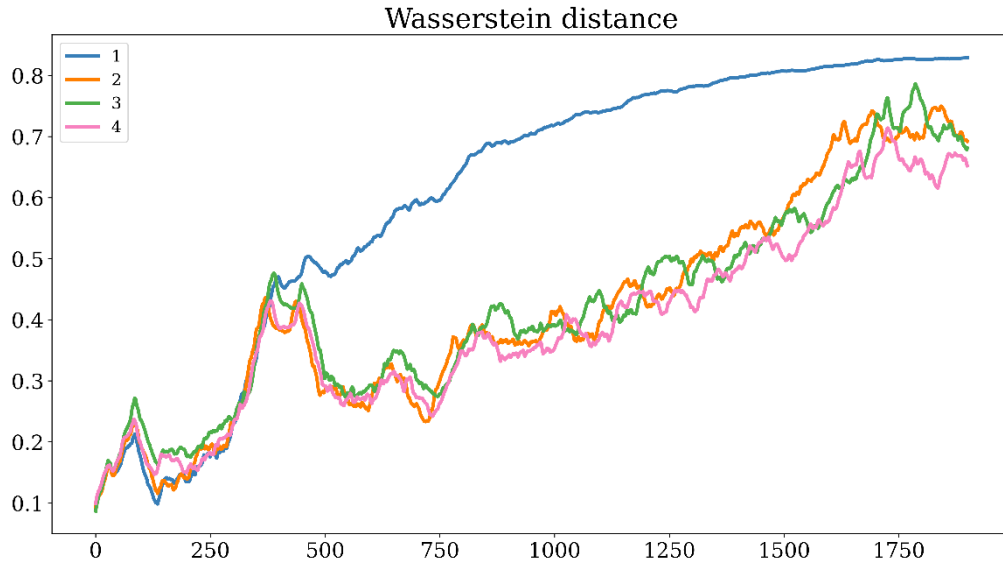


Figure 17. The Wasserstein distance for the data presented in Figure 15.

Precision is an important metric because large number of false positives reduce trustworthiness of the anomaly detection method. Therefore, the paired sample t-test and Kullback-Leibler divergence were the best of the tested methods, and the Kolmogorov-Smirnov test and the Wasserstein distance were not tested further.

The paired sample t-test and the Kullback-Leibler divergence had high F1-scores. F1-score is an important metric because it is the harmonic mean of precision and recall and therefore, it can be considered a measure of overall model performance. The Kullback-Leibler divergence scores for the data presented in Figure 15 are presented in Figure 18, and the paired sample t-test for the same data is presented in Figure 19.

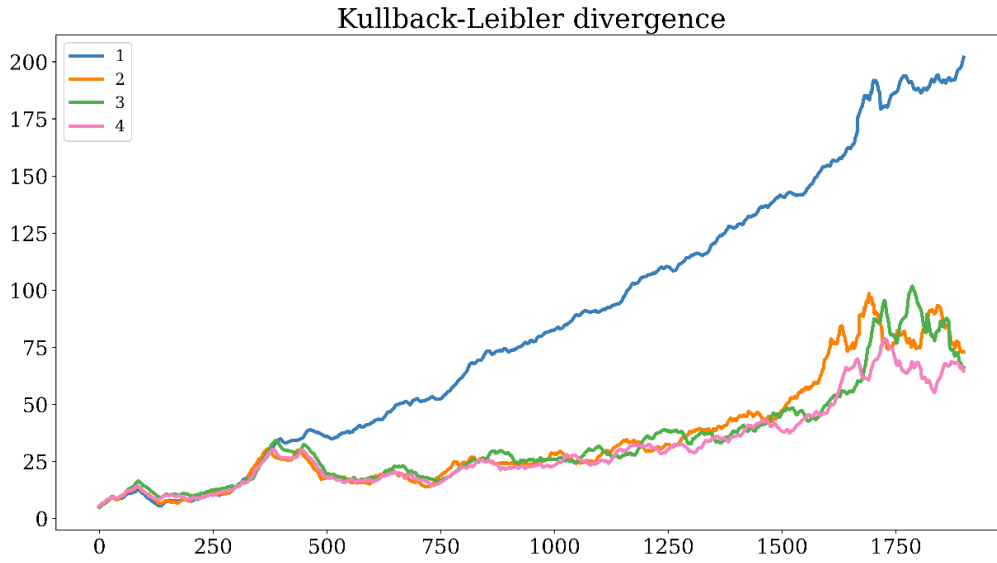


Figure 18. The Kullback-Leibler divergence scores for the data presented in Figure 15.

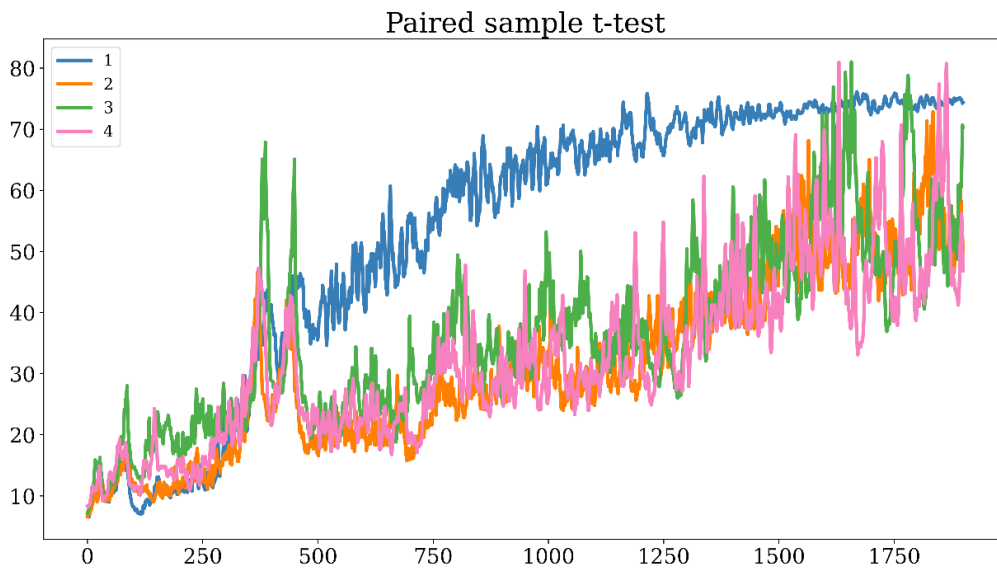


Figure 19. The paired sample t-test for the data presented in Figure 15.

## 4.2 Simulation testing

Ten data sets were selected for testing and drifts were induced using three different drift rates for both abrupt and gradual drifts. Therefore, the total number of simulated test cases was 30 for both short-term and long-term drifts. The results for long-term drift detection are presented in section 4.2.1 and the results for short-term drift detection are presented in section 4.2.2.

### 4.2.1 Long-term drift detection

The methods were tested using four different window sizes, which are 25, 50, 75 and 100. Results for the regression method are presented in Table 2. A small sliding window size did not work well for detecting slower drifts. Increasing the window size reduced the number of false negatives but the number of false positives remained quite large.

All drifts were induced at time 700 for all example data sets. Reaction times were estimated based on the difference between the drift induction and the detection time. For example, if the drift was detected at index 750, the drift was detected within 50 data points. The estimated detection time was 50 minutes because the sample rate is approximately one minute.

Using timestamps for evaluating the reaction time was not as accurate because there are gaps in the data. For example, in one example the drift was detected at time 719 which means that the drift was detected within 19 measurements, which should correspond to approximately 19 minutes. However, the reported reaction time was 2 hours and 9 minutes because there was a time gap between times 700 and 719. In another example, the drift was detected within 21 measurements, and the reaction time was 20 minutes. Therefore, estimated reaction times were reported instead of reaction times which were calculated based on the timestamps.

Table 2. Metrics for comparing performance of the regression method using different window sizes.

Window size	25	50	75	100
Accuracy	73.1	79.6	79.6	82.4
Precision	45.5	60.9	55.8	58.7
Recall	37.0	51.9	88.9	100.0
F1-score	40.8	56.0	68.6	74.0
Reaction time	2 hours 47 minutes	2 hour 43 minutes	2 hours 31 minutes	2 hours 33 minutes

Results using only the Kullback-Leibler divergence statistic are presented in Table 3. There were few false negatives, but there were many false positives. Results were similar when using the paired sample t-test and therefore, using only a statistic test was deemed non-effective for detecting anomalies reliably. Larger window sizes of 75 and 100 performed better compared to the sliding window sizes of 25 and 50.

Table 3. Metrics for comparing performance of the Kullback-Leibler divergence using different window sizes.

Window size	25	50	75	100
Accuracy	68.5	61.1	69.4	75.0
Precision	44.1	39.1	45.0	50.0
Recall	96.3	79.4	100.0	100.0
F1-score	60.5	52.4	62.1	66.7
Reaction time	3 hours 25 minutes	3 hours 15 minutes	2 hours 22 minutes	2 hours 34 minutes

The results for the combination of two of the previous methods are presented in Table 4. The Kullback-Leibler divergence was used as an additional test after comparing standardized slope differences. Drift was detected only if both the regression method and the Kullback-Leibler divergence detected a drift. The number of false positives was reduced without significantly increasing the reaction time. Therefore, the regression method with an additional statistical test was evaluated as the most prominent method among those tested.

Table 4. Metrics for comparing performance of regression method with Kullback-Leibler divergence using different window sizes.

Window size	25	50	75	100
Accuracy	88.0	92.6	97.2	97.2
Precision	71.9	88.0	90.0	90.0
Recall	85.2	81.5	100.0	100.0
F1-score	78.0	84.6	94.7	94.7
Reaction time	3 hours 25 minutes	3 hours 6 minutes	2 hours 53 minutes	2 hours 49 minutes

The Kullback-Leibler divergence score is calculated for each sensor using a sliding window. The latest calculated Kullback-Leibler divergence scores are stored in arrays for each sensor. The size of this array is half the size of the sliding window that was used to fit the regression line. These scores are

used to calculate the median and the median absolute deviation of the most recent Kullback-Leibler divergence scores for each sensor. A sensor is detected as anomalous if its modified z-score is larger than three when comparing it to the latest Kullback-Leibler divergence scores of the other sensors:

$$M_i = 0.6745 \frac{KL - \widetilde{KL}_i}{MAD_i}, \quad (21)$$

where  $KL$  is the Kullback-Leibler divergence score of the anomaly candidate sensor,  $\widetilde{KL}_i$  is the median Kullback-Leibler divergence score of the non-outlier sensor  $i$  and  $MAD_i$  is the median absolute deviation of the Kullback-Leibler divergence scores.  $KL$  for this data was calculated according to equation (20), where  $p_0$  is the recent window and  $p_1$  is the reference window that were formed using the same sliding window that was used for calculating the regression lines. Because division by zero is undefined,  $\varepsilon = 0.00001$  was added to values of the recent window  $p_0(x)$  and the reference window  $p_1(x)$  to avoid dividing by zero in this application. Sensor is detected as anomalous if  $|M_i| > 3$  for all non-outlier sensors  $i$ .

The results for the regression method with the paired sample t-test is presented in Table 5. It had a similar performance to the Kullback-Leibler method and therefore, both could be considered as suitable options for the considered application.

Table 5. Metrics for comparing performance of the regression method with the paired sample t-test using different window sizes.

Window size	25	50	75	100
Accuracy	92.6	93.5	93.5	97.2
Precision	100.0	85.7	81.3	90.0
Recall	70.4	88.9	96.3	100.0
F1-score	82.6	87.3	88.1	94.7
Reaction time	4 hours 42 minutes	3 hours 7 minutes	2 hours 47 minutes	2 hours 49 minutes

Including an additional statistical test improved the robustness of the regression-based drift detection method. Reaction times were quite slow for all methods. However, induced drifts were gradual long-term drifts that do not require as fast reaction times as abrupt sensor drifts, and they are harder to detect fast in a robust manner. Therefore, robustness of the method is more

important feature when it comes to analyzing the performance of long-term drift detection.

#### 4.2.2 Short-term drift detection

A drift detection method should be able to detect faster drifts in addition to more gradual drifts. Two separate window sizes were selected for detecting short-term and long-term drifts. Based on the results presented in Table 4 and Table 5, the combination of regression method and a selected statistic test, the Kullback-Leibler divergence or the paired sample t-test, were the best performing methods for detecting long-term drifts; for short-term drifts a similar approach was tested.

The Kullback-Leibler divergence scores for the data presented in Figure 14 are presented in Figure 20, and the paired sample t-test for the same data is presented in Figure 21. The difference between the anomalous sensor and other sensors was more distinct for the Kullback-Leibler divergence and therefore, it was easier to robustly detect anomalous sensor by comparing its test statistics to the other sensors. The anomalous sensor could also be clearly observed using the paired sample t-test statistics. However, the paired sample t-test was more prone to false positives because there was fluctuation in the test statistics of the other sensors and their values were closer to the anomalous sensor.

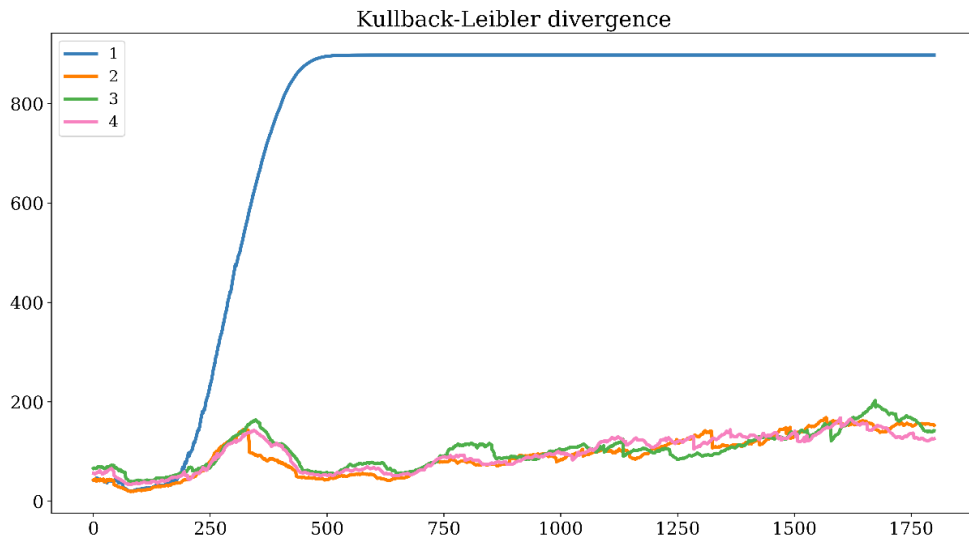


Figure 20. The Kullback-Leibler divergence scores for the short-term drift presented in Figure 14.

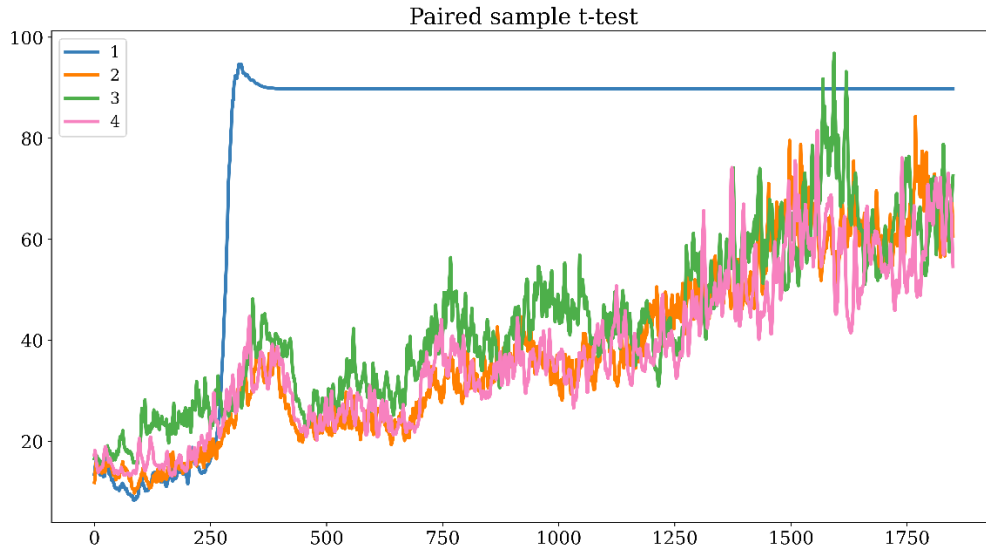


Figure 21. The paired sample t-test for the short-term drift presented in Figure 14.

The results for the regression method with the Kullback-Leibler divergence are presented in Table 6. Sliding window size of 5 was too small because the number of false negatives increased. Sliding window size of 10 had the overall best performance because the reaction time was the fastest and there were fewer false positives compared to larger sliding window sizes.

Table 6. Metrics for comparing performance of regression method with the Kullback-Leibler divergence for detecting short-term drifts.

Window size	5	10	15	20
Accuracy	91.7	94.4	88.9	91.7
Precision	75.0	88.9	69.2	75.0
Recall	100.0	88.9	100.0	100.0
F1-score	85.7	88.9	81.8	85.7
Reaction time	19 minutes	21 minutes	34 minutes	29 minutes

### 4.3 Case study

The number of real-world examples was limited but short-term and long-term drift detection were tested using a few real-world examples, which are presented in this section. The first example of a real-world data set that includes a drifting sensor is presented in Figure 22. Reaction time could be better, but a drift is detected in the correct sensor without any false positives.

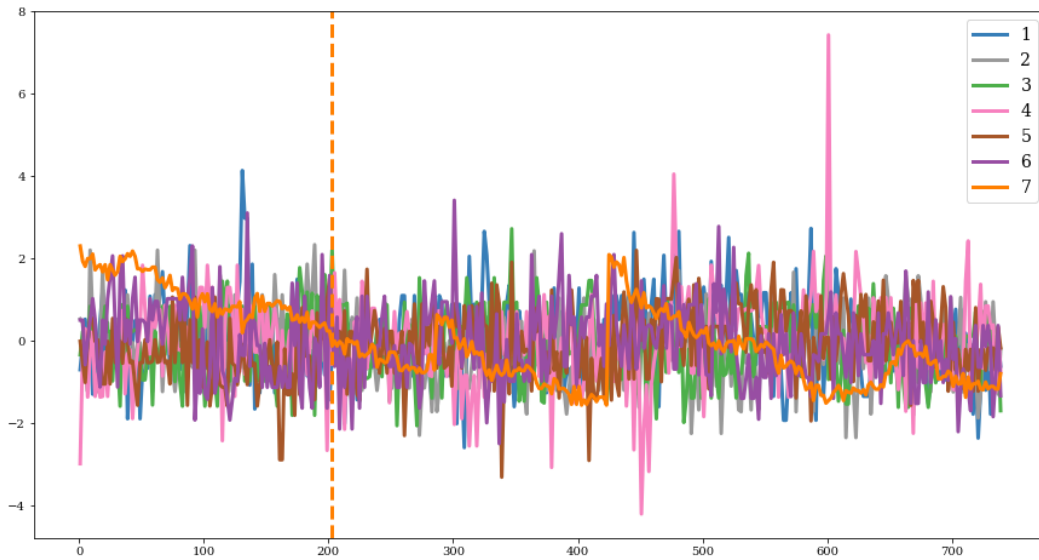


Figure 22. Example of detected drift in a real-world data set.

Another real-world example is presented in Figure 23. This is a more challenging example for the algorithm because more than one sensor is behaving unexpectedly, and there is fluctuation in the values of sensor 3. Drifts were not initially detected for this dataset. The main reason was that it was required that the regression slope of the anomalous sensor would have to be different from the regression slopes of at least two other sensors. Drifts were detected when this requirement was removed. The requirement that the regression slope of the outlier sensor must be different from at least two other sensors may make the algorithm too robust in some examples in which multiple sensors are behaving unexpectedly.

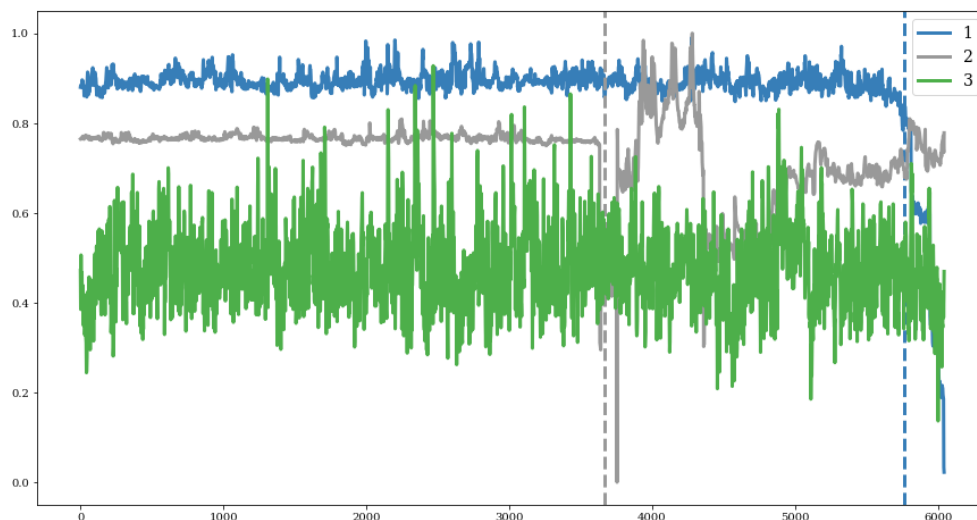


Figure 23. Example of detected drifts in a real-world data set.

A real-world example of abrupt drift is presented in Figure 24. Drift is detected in the correct sensor without false positives, and therefore, the algorithm is performing as expected.

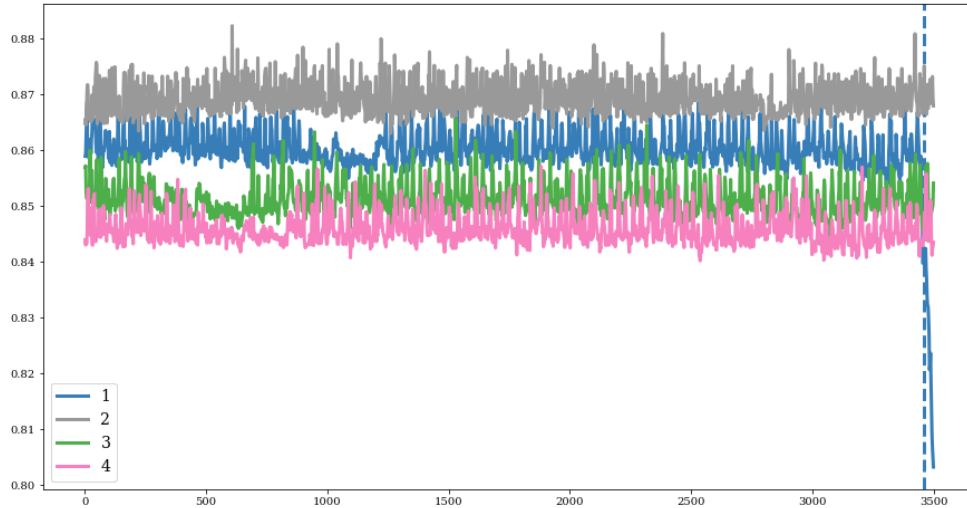


Figure 24. Example of detected abrupt drift in a real-world data set.

#### 4.4 Final method

The final method and its performance are described in this section. The best method was the regression-based method combined with the Kullback-Leibler divergence. The final version of the algorithm starts with the calculation of the Theil-Sen regression lines for each sensor. Two different window sizes are used, which are 10 for detecting short-term drifts, and 100 for detecting long-term drifts.

These regression lines are compared by calculating standardized slope differences between each sensor pair. Sensor pair is determined as a candidate anomaly if the absolute value of the standardized slope difference is larger than three. Outlier candidate is a sensor, which has the smallest slope of the pair, and all other sensors are marked as non-outliers.

The Kullback-Leibler divergence is calculated for each sensor using the most recent measurements after the regression line calculation. The size of this calculation window is the same that was used for calculating the regression line, and it is divided into reference and detection windows. The Kullback-Leibler divergence scores of the outlier sensor are compared to the scores of the non-outlier sensors as presented in equation (21).

Drift is detected if the regression method and the Kullback-Leibler divergence detect a sensor as an anomaly. The regression slope of this sensor must

be significantly different from at least two other sensors based on the standardized slope differences.

The method was tested in section 4.2 separately using gradual and abrupt drifts. However, drifts cannot be automatically classified as abrupt or gradual in the real world and thus, the algorithm should be able to detect different types of drifts simultaneously without any knowledge of the type of the drift. Therefore, the algorithm was also tested using a combination of short-term and long-term drifts.

Short-term drift detection worked as expected but the larger sliding window size of the long-term drift detection resulted in false positives in some data sets that contained abrupt drifts. Therefore, an additional regression line comparison was added to the gradual drift detection:

$$M_i = 0.6745 \frac{b - b_i}{MAD}, \quad (22)$$

where  $b$  is the regression slope of the anomaly candidate,  $b_i$  is the latest regression slope of the non-outlier sensor  $i$  and  $MAD$  is the median absolute deviation of the latest non-outlier regression slopes. Drift is detected if  $|M_i| > 3$  for all non-outlier sensors  $i$ . This additional comparison may increase the reaction time, but robustness of the long-term drift detection is improved.

Results for the final version of the drift detection method based on 216 tests are presented in Table 7. There were seven false positives and zero false negatives. The number of false positives was reduced to zero when the standardized slope difference threshold was raised from three to five. However, using larger threshold may increase the reaction time especially when detecting abrupt drifts.

Table 7. Metrics for comparing performance of final version of the drift detection method.

Standardized slope difference threshold	3	5
Accuracy	96.8	100.0
Precision	88.5	100.0
Recall	100	100.0
F1-score	93.9	100.0

Parameters of the final version of the algorithm are presented in Table 8. The parameters are mostly similar for short-term and long-term version of the algorithm. The algorithm is optimized to be as robust as possible with these parameters.

Table 8. Parameters of the final method.

	Short-term	Long-term
Window size	10	100
Regression slope confidence interval	95%	95%
Margin of error	3.92	3.92
Standardized slope difference threshold	5	5
The Kullback-Leibler divergence threshold	3	3
Long-term regression slope comparison threshold	3	3
Outlier count	5	7

Different parameters could be considered for the two different versions of the algorithm. For example, standardized slope difference threshold could be reduced to three for the short-term version of the algorithm so that the reaction time could be reduced.

The outlier count is the number of consecutive times a sensor must be detected as an anomaly before an alert is raised. This is another parameter that could be different for short-term and long-term versions of the algorithm. The outlier count could be increased for the long-term version of the algorithm because it would most likely improve robustness. Reducing the outlier count to five for the short-term version of the algorithm would most likely reduce the reaction time, but as a result, there could be an increased number of false positives.

It is possible to tune these parameters so that accuracy or reaction time of the drift detection method can be improved. Parameters that are presented in Table 8 were chosen in such a way that the robustness of the algorithm is as good as possible while maintaining sufficient reaction time.

## 5 Summary

The objective of this thesis was to develop an improved drift detection method for sensor data. The existing monitoring solution in this system was a user-defined threshold. Long-term drift detection was challenging using a static threshold and therefore, developing a more advanced drift detection method made it possible to detect gradual changes more reliably.

The considered sensors measure physical quantities, such as pressure and temperature, and they are located close to each other. Therefore, correlation of the sensors was the main idea behind different methods that were tested. A drift detection method should be able to detect if values of one sensor start to decrease faster compared to the other sensors. Two main requirements were robustness and responsiveness. Drifts should be detected reliably without too many false positives while maintaining sufficient reaction time.

The first method that was developed for this problem was a regression-based method. Regression lines were fitted separately to the last data points of each sensor. A sensor was classified as anomalous if its regression slope was significantly different from the regression slopes of the other sensors.

Regression lines quantified drifts in the sensor data quite well and therefore, regression-based method was easy to justify theoretically. One of the main challenges of this method was that there were many parameters, such as window size and anomaly score thresholds, and selecting optimal parameters was not straightforward. However, parameters that worked for different drifts could be chosen based on testing. The parameters can be easily adjusted if there are too many false positives or if the reaction times are too long based on further testing.

Two different window sizes were used for drift detection because both short-term and long-term drifts may occur in the sensors. A larger window size worked well for detecting gradual drifts, and a smaller window size worked well for detecting faster drifts. Selecting an optimal window size is important when it comes to improving reaction times and therefore, detecting optimal sliding window size based on the data is an important area for future research.

The second developed method was based on Bayesian autoencoder. An autoencoder is an artificial neural network that is used to learn representations of the original input data, and these representations are reconstructed. In the initialization phase, a Bayesian autoencoder was trained when the sensors were working as expected. After the initialization, data from the sensors was given as input to the autoencoder model. If there was no drift in the data, the reconstruction from the input data resembled the training data. However, if drift was present in the data, autoencoder was not able to reconstruct data, which caused the reconstruction loss to increase.

Reconstruction loss reacted quite well to drifts in the data. However, there were some test examples in which the behaviour of the reconstruction loss

was quite unexpected. An autoencoder is also a more computationally expensive method compared to the tested statistical methods, and this was one of the reasons why Bayesian autoencoder was not tested extensively.

The selected statistical tests were also tested for drift detection. Two of the best performing methods were the paired sample t-test and the Kullback–Leibler divergence. The Kullback-Leibler divergence test statistic for the anomalous sensor was more distinct based on visualizations and therefore, it was selected as the better statistical test for this problem. However, neither of these methods was able to detect the investigated drifts robustly without additional methods.

Thus, these statistical tests were tested with the regression-based method. One of the main findings was that adding a statistical test significantly improved robustness of the drift detection method. Neither the regression-based method nor a statistical test was able to detect drifts robustly, but a combination of these two methods was robust.

The best performing method was regression-based method combined with the Kullback-Leibler divergence. This method was able to detect gradual long-term drifts robustly and therefore, gradual drift detection was improved compared to the existing benchmark method that is based on user-defined threshold. However, the reaction times for short-term drift detection could be improved in the future. The regression-based method may react slower compared to some other methods because regression lines change more gradually compared to individual data points. The cumulative sum-based method is one possible method that could be tested in the future (Yi & Qiu, 2021, p. 892).

Accuracy or reaction time of the drift detection method can be improved by tuning the parameters. The parameters of the final method were optimized to make the method as robust as possible. False positives should be avoided because too many false positives reduce trustworthiness of the method. It is possible to improve the reaction time of this drift detection algorithm using other parameters, but it is important to consider the trade-off between accuracy and reaction time.

## References

- Albattah, A. & Rassam, M. A. 2022. A Correlation-Based Anomaly Detection Model for Wireless Body Area Networks Using Convolutional Long Short-Term Memory Neural Network. *Sensors*, vol. 22, no. 1951. Available at: <https://doi.org/10.3390/s22051951>.
- Aminikhanghahi, S. & Cook, D. J. 2017. A survey of methods for time series change point detection. *Knowledge and Information Systems*, vol. 51, pp. 339–367. Available at: <https://doi.org/10.1007/s10115-016-0987-z>.
- Ayadi, A., Ghorbel, O., Obeid, A. M. & Abid, M. 2017. Outlier detection approaches for wireless sensor networks: A survey. *Computer Networks*, vol. 129, pp. 319-333. Available at: <https://doi.org/10.1016/j.com-net.2017.10.007>.
- Baldewijns, G., Luca, S., Vanrumste, B. & Croonenborghs, T. 2016. Developing a system that can automatically detect health changes using transfer times of older adults. *BMC Medical Research Methodology*, vol. 16, no 23. Available at: <https://doi.org/10.1186/s12874-016-0124-4>.
- Bernardo, J. M., & Smith, A. F. M. 2009. Bayesian theory. *Wiley Series in Probability and Statistics*, vol. 405. John Wiley & Sons. 608 p. ISBN 047031771X.
- Bifet, A., & Gavaldà, R. 2007. Learning from time-changing data with adaptive windowing. In *Proceedings of the 2007 SIAM international conference on Data Mining*, pp. 443-448. Available at: <https://doi.org/10.1137/1.9781611972771.42>.
- Bosman, H. H. W. J., Iacca, G., Tejada, A., Wörtche, H. J. & Liotta, A. 2017. Spatial anomaly detection in sensor networks using neighborhood information. *Information Fusion*, vol. 33, pp. 41-56. Available at: <https://doi.org/10.1016/j.inffus.2016.04.007>.
- Bosman, H. H. W. J. 2016. Anomaly detection in networked embedded sensor systems. Doctoral dissertation. Eindhoven University of Technology. Eindhoven. ISBN: 978-90-386-4125-6.
- Braei, M. & Wagner, S. 2020. Anomaly Detection in Univariate Time-series: A Survey on the State-of-the-Art. Available at: <https://doi.org/10.48550/arXiv.2004.00433>.

Chander, B. & Kumaravelan, G. 2022. Outlier detection strategies for WSNs: A survey. *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, pp. 5684-5707. Available at: <https://doi.org/10.1016/j.jksuci.2021.02.012>.

Chandola, V., Banerjee, A. & Kumar, V. 2009. Anomaly detection: A survey. *ACM Computing Surveys*, vol. 41, no. 3, pp. 1-58. Available at: <https://doi.org/10.1145/1541880.1541882>.

Chandra, R., Jain, M., Maharana, M. & Krivitsky, P. N. 2022. Revisiting Bayesian Autoencoders with MCMC. *IEEE Access*, vol. 10, pp. 40482-40495. Available at: <https://doi.org/10.1109/ACCESS.2022.3163270>.

Chatterjee, A. & Ahmed, B. S. 2022. IoT anomaly detection methods and applications: A survey. *Internet of Things*, vol. 19, no. 100568. Available at: <https://doi.org/10.1016/j.iot.2022.100568>.

Cook, A. A., Misirli G. & Fan, Z. 2020. Anomaly Detection for IoT Time-Series Data: A Survey. *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6481-6494. Available at: <https://doi.org/10.1109/JIOT.2019.2958185>.

Erhan, L., Ndubuaku, M., Di Mauro, M., Song, W., Chen, M., Fortino, G., Bagdasar, O. & Liotta, A. 2021. Smart anomaly detection in sensor systems: A multi-perspective review. *Information Fusion*, vol. 67, pp. 64-79. Available at: <https://doi.org/10.1016/j.inffus.2020.10.001>.

Foorthuis, R. 2021. On the nature and types of anomalies: a review of deviations in data. *International Journal of Data Science and Analytics*, vol. 12, pp. 297-331. Available at: <https://doi.org/10.1007/s41060-021-00265-1>.

Fu, X., Wang, Y., Li, W., Yang, Y. & Postolache, O. Lightweight Fault Detection Strategy for Wireless Sensor Networks Based on Trend Correlation. *IEEE Access*, vol. 9, pp. 9073-9083. Available at: <https://doi.org/10.1109/ACCESS.2021.3049837>.

Higgins J. P. T., Li T., Deeks J. J (editors). Chapter 6: Choosing effect measures and computing estimates of effect. In: Higgins J. P. T., Thomas J., Chandler J., Cumpston M., Li T., Page M. J., Welch V. A. (editors). *Cochrane Handbook for Systematic Reviews of Interventions* version 6.3 (updated February 2022). Cochrane, 2022. Available from [www.training.cochrane.org/handbook](http://www.training.cochrane.org/handbook).

Hilal, W., Gadsden, S. A. & Yawney, J. 2022. Financial Fraud: A Review of Anomaly Detection Techniques and Recent Advances. *Expert Systems with*

Applications, vol. 193. Available at:  
<https://doi.org/10.1016/j.eswa.2021.116429>.

Jamshidi, E. J., Yusup, Y., Kayode, J. S. & Kamaruddin, M. A. 2022. Detecting outliers in a univariate time series dataset using unsupervised combined statistical methods: A case study on surface water temperature. *Ecological Informatics*, vol. 69, no. 101672. Available at:  
<https://doi.org/10.1016/j.ecoinf.2022.101672>.

Jinwon, A. & Sungzoon, C. 2015. Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE*, vol. 2, no. 1, pp. 1-18. Available at: <https://www.semanticscholar.org/paper/Variational-Autoencoder-based-Anomaly-Detection-An-Cho/061146b1d7938d7a8dae70e3531a00fceb3c78e8>.

Johnson, D. & Sinanović, S. 2001. Symmetrizing the Kullback-Leibler Distance. *IEEE Transactions on Information Theory*. Available at: <https://scholarship.rice.edu/handle/1911/19969>.

Kajmakovic, A., Diwold, K., Römer, K., Pestana, J. & Kajtazovic, N. Degradation Detection in a Redundant Sensor Architecture. *Sensors* 2022, vol. 22, no. 4649. Available at: <https://doi.org/10.3390/s22124649>.

Khraisat, A., Gondal, I., Vamplew, P. & Kamruzzaman, J. 2019. Survey of intrusion detection systems: techniques, datasets and challenges. *Cybersecurity*, vol. 2, no. 20. Available at: <https://doi.org/10.1186/s42400-019-0038-7>.

Klein, S. & Verbeke, M. 2020. An unsupervised methodology for online drift detection in multivariate industrial datasets. *International Conference on Data Mining Workshops*, pp. 392-399. Available at: <https://doi.org/10.1109/ICDMW51313.2020.00061>.

Kumar, D. P., Amgoth, T. & Annavarapu, C. S. R. 2019. Machine learning algorithms for wireless sensor networks: A survey. *Information Fusion*, vol. 49, pp. 1-25. Available at: <https://doi.org/10.1016/j.inffus.2018.09.013>.

Leys, C., Ley, C., Klein, O., Bernard, P. & Licata L. 2013. Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. *Journal of Experimental Social Psychology*, vol. 49, no. 4, pp. 764-766. Available at: <https://doi.org/10.1016/j.jesp.2013.03.013>.

Li, K., Gao, X., Fu, S., Diao, X., Ye, P. Xue, B., Yu, J. & Huang, Z. 2022. Robust outlier detection based on the changing rate of directed density ratio. *Expert*

Systems with Applications, vol. 207, no. 117988. Available at: <https://doi.org/10.1016/j.eswa.2022.117988>.

Maharana, K., Mondal, S. & Nemade, B. 2022. A review: Data pre-processing and data augmentation techniques. Global Transitions Proceedings, vol. 3, no. 1, pp. 91-99. ISSN 2666-285X. Available at: <https://doi.org/10.1016/j.gltp.2022.04.020>.

Maleki, S., Maleki, S. & Jennings, N. R. 2021. Unsupervised anomaly detection with LSTM autoencoders using statistical data-filtering. Applied Soft Computing, vol. 108, no. 107443. Available at: <https://doi.org/10.1016/j.asoc.2021.107443>.

Munirathinam, S. 2021. Drift Detection Analytics for IoT Sensors. Procedia Computer Science, vol. 180, pp. 903-912. Available at: <https://doi.org/10.1016/j.procs.2021.01.341>.

Muskulus, M. & Verduyn-Lunel, S. 2011. Wasserstein distances in the analysis of time series and dynamical systems. Physica D: Nonlinear Phenomena, vol. 240, no. 1, pp. 45-58. Available at: <https://doi.org/10.1016/j.physd.2010.08.005>.

Paternoster, R., Brame, R., Mazerolle, P. & Piquero, A. R. 1998. Using the Correct Statistical Test for the Equality of Regression Coefficients. Criminology, vol. 36, no. 4, pp. 859-866. Available at: <https://doi.org/10.1111/j.1745-9125.1998.tb01268.x>.

Pearce, T., Leibfried, F. & Brintrup, A. 2020. Uncertainty in Neural Networks: Approximately Bayesian Ensembling. Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, vol. 108, pp. 234-244 Available at: <https://proceedings.mlr.press/v108/pearce20a.html>.

Porwik, P. & Dadzie, B. M. 2022. Detection of data drift in a two-dimensional stream using the Kolmogorov-Smirnov test. Procedia Computer Science, vol. 207, pp. 168-175. Available at: <https://doi.org/10.1016/j.procs.2022.09.049>.

Pramudita, B. A., Sumanto, B., Setiawan, N. A. & Ardiyanto, I. 2019. Performance Enhancement of Complete Ensemble Empirical Mode Decomposition (CEEMD) - Independent Component Analysis (ICA) In Ocular Artifact Removal. 5th International Conference on Science and Technology (ICST), pp. 1-6. Available at: <https://doi.org/10.1109/ICST47872.2019.9166351>.

Qin, H., Zhan, X. & Zheng, Y. 2022. CSCAD: Correlation Structure-based Collective Anomaly Detection in Complex System. *IEEE Transactions on Knowledge and Data Engineering*. Available at: <https://doi.org/10.1109/TKDE.2022.3154166>.

Ramdas, A., Trillos, N.G. & Cuturi, M. 2017. On Wasserstein Two-Sample Testing and Related Families of Nonparametric Tests. *Entropy*, vol. 19, no. 2. Available at: <https://doi.org/10.3390/e19020047>.

Rietveld, T. & van Hout, R. 2017. The paired t test and beyond: Recommendations for testing the central tendencies of two paired samples in research on speech, language and hearing pathology. *Journal of Communication Disorders*, vol. 69, pp. 44-57. Available at: <https://doi.org/10.1016/j.jcomdis.2017.07.002>.

Samara, M.A., Bennis, I., Abouaissa, A. & Lorenz, P. 2022. A Survey of Outlier Detection Techniques in IoT: Review and Classification. *Journal of Sensor and Actuator Networks*, vol. 11, no. 4. Available at: <https://doi.org/10.3390/jsan11010004>.

Sen, P. K. 1968. Estimates of the regression coefficient based on Kendall's tau. *Journal of the American Statistical Association*, vol. 63, no. 324, pp. 1379-1389. Available at: <https://doi.org/10.1080/01621459.1968.10480934>.

Sgueglia, A., Di Sorbo, A., Visaggio, C. A. & Canfora, G. 2022. A systematic literature review of IoT time series anomaly detection solutions. *Future Generation Computer Systems*, vol. 134, pp. 170-186. Available at: <https://doi.org/10.1016/j.future.2022.04.005>.

Theil, H. 1950. A rank-invariant method of linear and polynomial regression analysis. *Indagationes mathematicae*, vol. 12, no. 85.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P. & SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, vol. 17, no. 3, pp. 261-272. Available at: <https://doi.org/10.1038/s41592-019-0686-2>.

Wu, W., Cheng, X., Ding, M., Xing, K., Liu, F. & Deng, P. 2007. Localized Outlying and Boundary Data Detection in Sensor Networks. *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 8, pp. 1145-1157. Available at: <https://doi.org/10.1109/TKDE.2007.1067>.

Yi, F. & Qiu, P. 2021. An adaptive CUSUM chart for drift detection. *Quality and Reliability Engineering International*, vol. 38, no. 2. Available at: <https://doi.org/10.1002/qre.3020>.

Yong, B. X., Fathy, Y. & Brintrup, A. 2020. Bayesian Autoencoders for Drift Detection in Industrial Environments. *IEEE International Workshop on Metrology for Industry 4.0 & IoT*, pp. 627-631. Available at: <https://doi.org/10.1109/MetroInd4.0IoT48571.2020.9138306>.

Yong, B. X. & Brintrup, A. 2022. Bayesian autoencoders with uncertainty quantification: Towards trustworthy anomaly detection. *Expert Systems with Applications*, vol. 209, no. 118196. Available at: <https://doi.org/10.48550/arXiv.2202.12653>.

Yu, K., Shi, W. & Santoro, N. 2020. Designing a Streaming Algorithm for Outlier Detection in Data Mining — An Incremental Approach. *Sensors*, vol. 20, no. 5. Available at: <https://doi.org/10.3390/s20051261>.

Zhang, Y., Hamm, N. A. S., Meratnia, N., Stein, A., van de Voort, M. & Havinga P. J. M. 2012. Statistics-based outlier detection for wireless sensor networks. *International Journal of Geographical Information Science*, vol. 26, no. 8, pp. 1373-1392. Available at: <https://doi.org/10.1080/13658816.2012.654493>.

Zhao, P., Kurihara, M., Tanaka, J., Noda, T., Chikuma, S. & Suzuki, T. 2017. Advanced correlation-based anomaly detection method for predictive maintenance. *IEEE International Conference on Prognostics and Health Management (ICPHM)*, pp. 78-83. Available at: <https://doi.org/10.1109/ICPHM.2017.7998309>.