Data-Driven Methods in Pipeline Leakage Detection

by

Iman Amini

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Control Systems

Department of Electrical and Computer Engineering

University of Alberta

Abstract

Nowadays, leakage detection is of great importance as pipelines are the major means of transporting hydrocarbon fluids and gases. In this thesis, we propose two methods based on supervised learning and filtering to deal with the pipeline leakage detection problem.

First, a novel two-stage detection method is introduced to differentiate normal, leakage and transient conditions of pipelines. In this method, feature vectors are constructed from the flow rate and pressure using leakage characteristics. An artificial neural network (ANN) is used in the first stage of the detection to differentiate normal and abnormal conditions with the feature vectors as the inputs. In the second detection stage, simple logic is used to distinguish leakage and transient for data under abnormal condition. The method has been shown to have higher detection performance and fewer false alarms in comparison with the line balance and Kantorovich distance methods.

As the pipeline leak data is not always abundant to train supervised learning models, a filter-based method is proposed to detect pipeline leakage that does not require prior leak data for training. Based on studies in field data, we model pipeline leakage as an increase in the mean value of the flow rate difference between the inlet and the outlet sensors, where the increased value is unknown and subject to change. Then, an adaptive filter is proposed based on the estimated cumulative distribution function (CDF) of the data in steady-state condition using kernel density estimation. The proposed filter has better performance in small leaks in comparison with different benchmarks.

Preface

Chapter 3 and parts of Chapters 1 and 2 are presented in the conference paper "A Two-Stage Deep-Learning Based Detection Method for Pipeline Leakage and Transient Conditions". The co-authors include I. Amini, Y. Jing, T. Chen, A. Colin and G. Meyer. The paper is accepted and virtually presented at Electric Power and Energy Conference (EPEC) 2020 in Edmonton.

Also, a revised version of Chapter 4 and parts of Chapters 1 and 2 are intended to be submitted as a journal paper in the future.

Acknowledgements

First, I would like to express my deep gratitude to my co-supervisors, Dr. Tongwen Chen and Dr. Yindi Jing, for their continuous support, immense encouragement, insightful comments and challenging questions during my master's program. I find myself lucky to study master's program under the supervision of these great supervisors. Under their supervision, I have learned how to better organize my thoughts and ideas, break the problems into simple tasks and carry out each at once.

Second, I would like to graciously thank our industrial partner, Suncor Energy, for their support. Special thanks to Amanda Colin and Gordon Meyer from Suncor Pipeline for their generous support, thoughtful comments and helpful discussions in the course of completing the project. The completion of this work would be impossible without their collaboration. I would like to also thank Dr. Sirish L. Shah, University of Alberta emeritus professor, for his thoughtful comments during the steering committee meetings. In addition, I would like to thank the Natural Sciences and Engineering Research Council (NSERC) for the provision of financial support for the project.

Further, I would like to thank all of my colleagues in Dr. Chen's and Dr. Jing's groups. Special thanks to Hossein Roohi, Xing Xiong, Jun Shang, Wenkai Hu, Boyuan Zhou, Rezwan Parwez, Hanieh Seyed Alinezhad, Lily Wang, Junyi Yang, Jing Zhou, Mani Hemanth, Hao Yu and Harikrishna Rao for their support and kindness.

Last but not least, I cannot be more grateful to my parents and my brother, Amin, for their unconditional love and support. Also, I would like to thank all my friends who made my life happier and helped me to become a better person.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Alarm systems are crucial for the monitoring and controlling industrial plants, which trigger alarm messages to operators in the case of faults in the process [1]. The faults that an alarm system fails to report to the operators are referred to as missed alarms. Missed alarms can prevent operators from taking actions during faults in operation, which can cause catastrophes in terms of process operation and safety. On the other hand, if an alarm is triggered during normal operation, it is known as a false alarm. A high false alarm rate can decrease the operators' trust in the alarm system, which can cause dangers in the case of real crisis [2]. Optimal designs and analysis of the alarm systems have attracted great attention in the literature [3]–[7]. In alarm systems, the fault detection and isolation (FDI) system plays an important role in triggering the alarms and sending the alarm messages to the operators. There are several methods to reduce false alarms in FDI systems for uni-variate signals such as alarm optimal threshold tuning [8], delay timers [9], deadbands [10] and filters [11]–[16]. Pipeline leakage detection systems are a special type of alarm systems with the functionality of online monitoring of oil and gas pipeline operation and reporting possible leakage to operators.

In this chapter, pipeline leak detection systems and filter-based alarm systems are reviewed in Sections 1.1 and 1.2 respectively. Afterward, the thesis

objectives are given in Section 1.3. Finally, the thesis outline is provided in Section 1.4.

1.1 Pipeline Leak Detection

Pipelines play a pivotal role in transporting hydrocarbon products such as fossil fuels, gas, and other chemicals [17]. Although they are recognized as the safest mean of transporting hydrocarbon products, leakage occurrence can affect their reliability [18]. Often, existing pipelines are exposed to leakage due to the aging, corrosion, and damage by a third party. Leakage poses severe damages to the economy, environment and human safety such as energy waste and pollution [19]. Also, owners of pipelines are responsible for environmental clean-up and compensation of damages, which lead to a huge financial burden to the company [20]. Therefore, early leak detection and localization are highly desirable to help prevent the occurrence of crucial situations.

Several methods have been proposed to detect leakage in the pipelines. Generally, these methods fall into three categories: non-technical methods, hardware-based methods, and software-based methods [18].

- Non-technical methods include trained dogs and expert inspectors to detect and track the pipeline leakage. Remote monitoring is being carried out by modern technologies such as autonomous underwater vehicles (AUVs) [21] and drones [22] in the oil and gas industries. Although these methods are shown to be reliable and accurate, they depend on the existence of trained dogs, expert personnel or smart facilities which are not cost-efficient and applicable [20].

- Hardware-based methods generally utilize sensors on the external part of the pipeline to measure a specific variable in order to detect leakage occurrence [23]. The acoustic method [24], optical fiber [25], tracer

method and soil monitoring [26] are examples of hardware-based methods. The methods of this type are expensive and difficult to apply to existing pipelines. Therefore, they cannot be used as the main strategy for online monitoring of the pipelines [20].

- Software-based methods use fluid measurements such as the flow rate, pressure, temperature and other parameters to monitor discrepancies in online data [23]. The line balance approach is based on the difference between the inlet and outlet flow rate [27]. This method is cost-efficient and can be easily implemented; however, the threshold tuning can be challenging due to a strict trade-off between false alarms and detection. The pressure point analysis method monitors the statistical properties of the pressure before and after leak occurrence [28]. The drawback of this method is the triggering of false alarms during transient conditions. A novel method based on Kantorovich distance was proposed in [20], which was shown to be sensitive to changes in the pressure and flow rate difference. Pipeline modeling can also be used to detect leakage based on the difference between the predicted values and the measured values. This method, however, is computationally intensive and any inaccuracy of the models can largely affect its performance.

In recent years, a few methods that utilize advanced signal processing tools have also been proposed. They consider feature extraction of leakage data in the pressure and flow rate to perform the classification between normal and leakage conditions. In [29], an artificial neural network (ANN) classification method was proposed using the inlet and the outlet pressures and outlet flow rate as inputs. A machine learning based method was proposed in [30], which uses statistical and Markov-chain based features. In [31], wavelet and statistical features were fed to a multi-layer perceptron neural network classifier to

perform detection and localization. For the work in [32], the Pearson correlation coefficient was used to extract features from pipeline leakage data and a long-short term memory (LSTM) classifier was proposed for detection. The major challenge of methods using signal processing is high false alarm rates due to operational changes (e.g., transient conditions) or noisy data. These false alarms can overwhelm operators during online monitoring.

1.2 Filter-Based Alarm Systems

An alarm filter is one of the common methods to detect abnormalities in the uni-variate process data with the existence of noise. An alarm filter is an algorithm applied on a group of process data points and its output is compared to a threshold to trigger an alarm. Compared to the use of unfiltered univariate signal, filtering is a powerful tool in the removal of false alarms in noisy process data. The design of a proper filter depends on the distribution of the data in normal and abnormal conditions.

The design of the optimal filter requires probability density functions (PDFs) of normal and abnormal conditions. Depending on the properties of the PDFs, the complexity of the optimal filter is often high. Thus, it is not a common choice for industrial systems [13]. The moving average filter is the most common one in the industry due to its simplicity and good performance. It was shown in [14] that if the PDFs of both normal and abnormal conditions are symmetric and log-concave, the moving average filter is the optimal finite impulse response (FIR) filter in the sense of minimizing the weighted sum of the false alarm rate (FAR) and the missed alarm rate (MAR). There are several other filtering methods proposed in the literature. The median filter was designed in [2] and industrial data were used to show the effectiveness of the filter. Ranked order filters were applied on process data in [11] and the analytical relationship between the order of the filter and detection delay was

derived. The optimal linear filter was derived using differential evolution algorithm in [13] and its result was compared to the moving average and the general optimal filters.

1.3 Thesis Objective

In this thesis, targeting at alleviating false alarms caused by transient conditions as well as accurate leak detection, we propose a simple two-stage deep-learning based detection method to differentiate normal (steady-state), transient and leakage conditions in pipelines with fluid motion. This method utilizes the mean of the flow rate difference and the variation of the pressure as the features for each time-window. In order to increase the sensitivity to small changes and remove the effect of outliers in the classification, a modified tangent-hyperbolic estimator is used to normalize the features. In the first stage of the detection, a trained deep-learning-based classifier categorizes time-windows as normal or abnormal (including leakage or transient) conditions. In the second stage, leakage and transient conditions are separated using simple logic. In addition, leak size estimation and size-restriction are used to remove false alarms caused by noises and disturbances. The proposed method is tested using industry data and is shown to have a high F1 score and a low FAR. Test results also show that the proposed method leads to a better overall performance in comparison to two methods in the literature.

Moreover, as industrial pipelines operate in the normal condition for the majority of their operation time, there is an imbalance between recorded leakage and normal condition data. In addition, there are some pipelines in the industry with no real or experimental leakage records. Therefore, the use of supervised learning is not always possible. With that consideration, a novel adaptive filter based on the naive Bayes classifier and a kernel density estimation is proposed to deal with the leakage problem without the usage of

historical leakage data for training. First, we model pipeline leakage as an increase in the mean value of the flow rate difference data, where the increased value is unknown and changing. The data points are segmented into time-windows. With the assumption that the data points in each time-window are independent and the idea of the naive Bayes classifier, we design a filter based on the cumulative distribution function (CDF) of the data in the steady-state condition and an estimated minimum change in the mean value in the positive direction when abnormality happens. As there is no prior assumption on the shape of the CDF of the data in the normal condition, a kernel density estimation is used to estimate the CDF based on the data in an adaptive mode. This filter is tested using simulated data under different probability density functions (PDFs). The results show that the proposed method has a better performance in terms of the receiver operating characteristic (ROC) curve compared to some benchmarks. Also, the proposed filter is tested on the fault scenarios with different signal-to-noise ratio (SNR) values as defined in this thesis. The result showed that the proposed method has a higher detection rate (DR) in comparison with benchmarks at the false alarm rate (FAR) of 0.1 in the case of low SNRs for data with different PDFs. The method is applied to real pipeline detection by adding an upper threshold on the inlet pressure difference signal to distinguish leakage from the step-up (transient) condition. The proposed algorithm is tested using three industrial datasets and is shown to have better overall performance in the detection of small leakage scenarios.

1.4 Thesis Outline

The rest of the thesis includes three chapters as follows.

- Chapter 2 includes some mathematical background required to understand the proposed algorithms and the performance evaluation in Chap-

ters 3 and 4. Pipeline leak signature and the difference of normal (steady-state), leakage and step-up and step-down transient conditions are discussed in Section 2.1. In Section 2.2, the kernel density estimation method is explained and the kernel and bandwidth selections are discussed. In section 2.3, the fault detection problem is associated with the classification problem and the performance measures to evaluate different methods throughout the thesis are introduced. In Section 2.4, the naive Bayes classifier is briefly explained. Finally, common detection filters are included in Section 2.5, including the log-likelihood ratio (LLR) filter, the moving average filter and the median filter.

- In Chapter 3, a novel two-stage scheme based on artificial neural networks (ANNs) and simple logic is proposed to categorize normal (steady-state), leakage and transient conditions. In Section 3.1, the system description and the problem statement are given. The proposed algorithm, including data pre-processing, stage 1 detection and stage 2 detection, is discussed in Section 3.2. Finally, the performance validation and the conclusion are given in Sections 3.3 and 3.4.

- Chapter 4 consists of a novel adaptive naive Bayes classifier based filter using the kernel density estimation. First, the problem model and assumptions are stated. In Section 4.1, the pipeline leakage model is modeled as the change in the mean value of the flow rate difference data in the positive direction. The issues of applying the optimal LLR filter to the problem are mentioned in section 4.2. The proposed method and leak detection scheme are discussed in Section 4.3. Section 4.4 defines the SNR measure for this problem and discusses the implementation requirements of different filters. Section 4.5 contains the comparison between the proposed method and some benchmarks using simulated and

real industrial data. Finally, Section 4.6 concludes Chapter 4.

- In Chapter 5, conclusion and future work for this thesis are provided. In Section 5.1, the overall conclusion of the thesis is provided. In Section 5.2, the future work to improve the proposed methods in Chapters 3 and 4 are suggested.

Chapter 2

Background Material

In this chapter, some mathematical background and common knowledge used in the system modeling, algorithms and performance validation of the thesis are provided. First, the pipeline characteristics during normal, leakage and transient conditions are compared. Then, the kernel density estimation method to estimate the PDF and CDF of a set of samples from a distribution is introduced. In the next part, the problem of fault detection is associated with a binary classification problem and the metrics to assess the performance of a fault detection method are reviewed. Finally, the naive Bayes classifier and detection filters are explained in the last two sections of this chapter.

2.1 Pipeline Leak Signature

Pipeline leakage has two significant characteristics known as "leak signatures", exposing the leakage in pressure and flow rates [20]. The first one is that the pressure drops at all measurement nodes [20]. The second behavior is based on mass conservation, which states that the fluid remains inside the pipeline until it exits from the ending node [23]. During the steady-state condition, the inlet and outlet volume flow rates inside the pipeline are balanced, provided that the fluid density and cross-sectional area of the pipeline remain constants along the pipeline. Therefore, the theoretical relationship between the inlet

and outlet nominal flow rates is given as follows:

$$\dot{V}_i(t) - \dot{V}_o(t) = 0,$$

where $\dot{V}_i$ and $\dot{V}_o$ are the inlet and the outlet volume flow rates respectively. On the other hand, when a leakage happens, due to fluid loss in the pipeline, the outlet flow rate is less than the inlet flow rate, which causes an imbalance between the inlet and the outlet flow rates [23]. That is,

$$\dot{V}_i(t) - \dot{V}_o(t) > 0.$$

The inlet-outlet flow rate difference has the opposite sign to the pressure change for a leakage.

On the other hand, during pipeline transient conditions, the pressure and the flow rate data can either increase or decrease, depending upon the changes made by the pipeline control center operator. This can cause extra difficulty in the leak detection design as it is hard to differentiate a leak from a transient condition. For instance, when there is a step-down in the pipeline, the pressure decreases, which is the same as the first characteristic of the leakage. Also, during other kinds of transient conditions, e.g., a step-up (causing an increase in the flow rate), in the pipeline, imbalance in the flow difference is positive, which has the same trend as a leakage. Nevertheless, there are signatures to separate leakage and transient conditions. For the first example above, the flow rate difference is negative, which is different from the leak signature. For the second transient condition, the pressure increases, which is different from a leakage. Therefore, both the flow rate difference and pressure signals are required to distinguish leak from transient conditions [20]. Fig. 2.1 shows the flow rate difference between the inlet and the outlet sensors and the inlet pressure signals for both process step-up, process step-down and leakage.

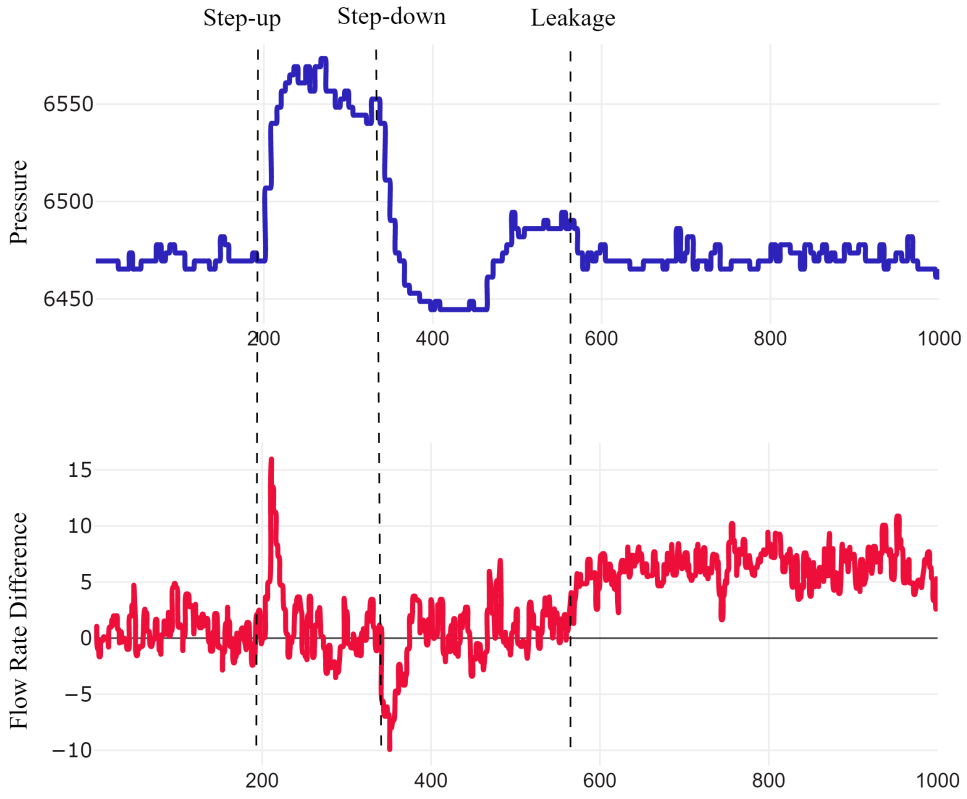

Figure 2.1: Flow rate difference and inlet pressure data during leakage and transient conditions (scaled industrial data).

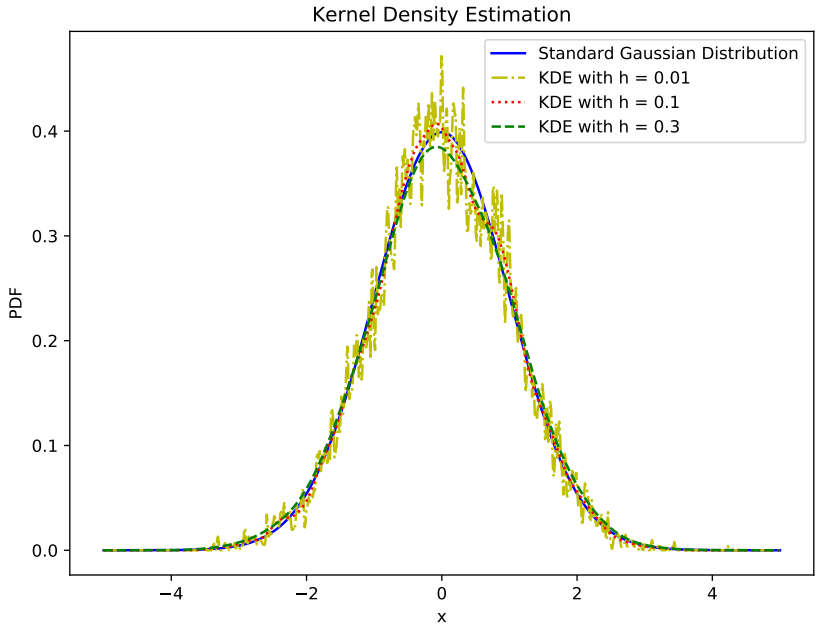

Figure 2.2: Kernel density estimation of the standard Gaussian distribution with different bandwidth values using 1000 points.

2.2 Kernel Density Estimation

The kernel density estimation is one of the non-parametric approaches to estimate the PDF of a random variable using sample data. Under the assumption that $x_1, x_2, ..., x_n$ are samples of a random variable X, the estimated PDF is obtained as below:

$$\widehat{f}_h(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right),$$

where K is a non-negative function known as the kernel function, and h is a positive value referred to as the bandwidth parameter [33]. The bandwidth parameter plays a role in controlling the smoothness of the estimation [34]. Fig. 2.2 depicts the kernel density estimation curves using different bandwidth values. As it can be observed, using larger bandwidth parameters can increase the smoothness of the estimation curve. However, it can affect the accuracy of the estimation in terms of the mean squared error.

Different kernel functions such as uniform, Epanechnikov, Gaussian, and others are utilized to estimate the PDF. One of the most common functions, known as Gaussian Kernel, is defined as below:

$$\varphi(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2}.$$

Among different kernel functions, the Epanechnikov kernel is indicated to be optimal in terms of minimizing mean square error [35]. However, the choice of the kernel has little effect on the accuracy of the estimation. Indeed, the bandwidth choice is of more importance as it establishes a trade-off between accuracy and smoothness [34].

There are several methods for bandwidth selection [36]. In this work, we use one of the most simple ones in order to achieve a balance between accuracy and algorithm execution time. Using the Gaussian Kernel function, the optimal bandwidth for underlying Gaussian distributions can be estimated as below [37]:

$$h = 1.06 \min\left(\hat{\sigma}, \frac{R}{1.34}\right) n^{-1/5},$$

where $\hat{\sigma}$ is the standard deviation of the samples and R is the interquartile range for the distribution. With the Gaussian kernel and the bandwidth rule, the estimated CDF using kernel density estimation is obtained as below:

$$\begin{aligned}
\hat{F}_{\mathrm{h}}(x) &= \int_{\infty}^{x} \frac{1}{nh} \sum_{i=1}^{n} \varphi\left(\frac{t - x_i}{h}\right) dt \\
&= \frac{1}{nh} \sum_{i=1}^{n} \int_{\infty}^{x} \varphi\left(\frac{t - x_i}{h}\right) dt \\
&= \frac{1}{n} \sum_{i=1}^{n} \phi\left(\frac{x - x_i}{h}\right)
\end{aligned}$$

where $\Phi(.)$ is the CDF of the standard normal distribution, defined by the following integral form:

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-t^2/2} dt.$$

2.3 Fault Detection and Performance Measures

The goal of fault detection is to trigger an alarm in the case of detecting an abnormality in the data. Fault detection can be associated with a binary classification problem.

First, let us explain the classification problem. The classification problem is to find a categorical output y for inputs $\mathbf{x}$, where $y \in \{1, 2, ..., C\}$ and C is known as the number of classes. If $C = 2$, the problem is called a binary classification problem. One method to solve the classification problem is the maximum a posteriori probability rule as below:

$$\hat{y} = \arg\max_{c} P(y = c \mid \mathbf{x}, \mathcal{D})$$

where $\mathcal{D}$ is the training dataset, $\hat{y}$ is the classification output and $P(y = c \mid \mathbf{x}, \mathcal{D})$ is the probability of class c given the input $\mathbf{x}$ and the training set $\mathcal{D}$. [38].

In the fault detection problem, we have two classes known as normal and abnormal classes. In this thesis, we segment time-series data into time-windows. Therefore, each time-window is classified or detected as normal or abnormal.

There are many widely used measures to evaluate the performance of detection methods. Here, we explain some of the most common methods. Traditionally, alarm triggering is associated with "positive" and "negative" classes. The events of true positive (TP), false positive (FP), true negative (TN) and false negative (FN) are defined as below.

- TP: true detection of an abnormal condition.
- FP: false detection of a normal condition. It is also referred to as error type I.
- TN: true classification of a normal condition.

- FN: missed detection of an abnormal condition. It is also referred to as error type II.

Therefore, the notions of false alarm rate (FAR), missed alarm rate (MAR) and detection rate (DR) are defined as below:

$$FAR = \frac{\text{number of } FP}{\text{number of } FP + \text{number of } TN},$$

$$MAR = \frac{\text{number of } FN}{\text{number of } TP + \text{number of } FN},$$

$$DR = \frac{\text{number of } TP}{\text{number of } TP + \text{number of } FN}.$$

where DR is also known as the probability of detection (PD). Also, the DR versus FAR curve is referred to as the receiver operating characteristic (ROC) curve.

In addition, the F_1 score is another useful measure, which is defined as

$$F_1 = \frac{2PR}{P + R},$$

where P and R, called the precision and recall rates respectively, are defined as follows:

$$P = \frac{\text{number of } TP}{\text{number of } TP + \text{number of } FP},$$
$$R = \frac{\text{number of } TP}{\text{number of } TP + \text{number of } FN}.$$

Finally, the detection delay (DD) is considered as the difference between the index of the detected time-window and the index of the starting faulty time-window.

2.4 Naive Bayes Classifier

The naive Bayes classifier is one of the machine learning methods to perform classification, under the independency assumption on the features of the input

data [39]. Let us assume the binary classification problem. The conditional probability of class c given the feature vector $\mathbf{x}_{nb}$ containing n features is obtained based on the Bayes' theorem as below:

$$P(c|\mathbf{x}_{nb}) = \frac{P(c)P(\mathbf{x}_{nb}|c)}{P(\mathbf{x}_{nb})}.$$

By taking the independence of features into account, $P(c|\mathbf{x}_{nb})$ is proportional to the following expression:

$$P(c|\mathbf{x}_{nb}) \propto P(c)\prod_{i=1}^{n} P(\mathbf{x}_{nb}^{i}|c),$$

where $\mathbf{x}_{nb}^{i}$ is the ith feature of $\mathbf{x}_{nb}$. Therefore, the classification is performed using the maximum a posteriori probability rule as follows [40]:

$$\hat{y} = \arg\max_{c} P(c)\prod_{i=1}^{n} P(\mathbf{x}_{nb}^{i}|c).$$

2.5 Detection Filters

In this section, we explain the general optimal filter to perform the detection given the PDFs of normal and abnormal condition. Also, we introduce some common practical filtering methods. First, denote $x[m]$ as the sample value of signal x at the time m. Denote the PDFs of $x[m]$ in the normal and abnormal conditions as $f_n(x)$ and $f_{ab}(x)$ respectively, The kth window with N sample points is obtained as below:

$$\mathbf{x}[k] = [x[k-N+1], x[k-N+2], ..., x[k]]\,.$$

The optimal filter and the optimal trip point (threshold), denoted by $y[k] = g(\mathbf{x}[k])$ and y_{tp} respectively, are obtained from the solution of the following optimization problem with no constraints:

$$\arg\min_{f, y_{tp}} J(g),$$

$$\text{where, } J(g) = c_1 \int_{y_{tp}}^{+\infty} f_{Y_n}(y_o)\mathrm{d}y_o + c_2 \int_{-\infty}^{y_{tp}} f_{Y_{ab}}(y_o)\mathrm{d}y_o,$$

$f_{Y_n}(y_o)$ and $f_{Y_{ab}}(y_o)$ are the PDFs of $y_o[k]$ under normal and abnormal conditions. The two terms of the objective function are the FAR and MAR. That is,

$$\text{FAR} = \int_{y_{tp}}^{+\infty} f_{Y_n}(y_o)\mathrm{d}y_o,$$

$$\text{MAR} = \int_{-\infty}^{y_{tp}} f_{Y_{ab}}(y_o)\mathrm{d}y_o.$$

Under the assumption of $c_1 = c_2$, the solution of the optimization problem has been shown as follows [13]:

$$y_o[k] = \sum_{m=k-N+1}^{k} \ln \frac{f_{ab}(x[m])}{f_n(x[m])}, \tag{2.1}$$

$$y_{tp} = \ln\left(\frac{c_1}{c_2}\right) = 0. \tag{2.2}$$

The filter as defined in Eq. (2.1) is also known as the LLR filter.

The classification task with labels "0" and "1", respectively, is conducted through the threshold function as below:

$$\delta_{y_o}(k) = \begin{cases} 1, & \text{if } y_o[k] \geq y_{tp} \\ 0, & \text{if } y_o[k] < y_{tp}. \end{cases}$$

If $G(\cdot)$ is a monotonic function, it holds that:

$$y_o[k] \geq \ y_{tp} \Leftrightarrow \begin{cases} G(y[k]) \geq G(y_{tp}), & \text{if } G(.) \text{ is an increasing function} \\ G(y[k]) \leq G(y_{tp}), & \text{if } G(.) \text{ is a decreasing function.} \end{cases}$$

Therefore, any scaling with a constant coefficient or shifting does not affect the optimality of the filter.

In addition, there are simpler filtering methods such as moving average and median filters. These methods use a threshold on the average and median values of the time-window $\mathbf{x}[k]$ to perform the detection. The expressions for

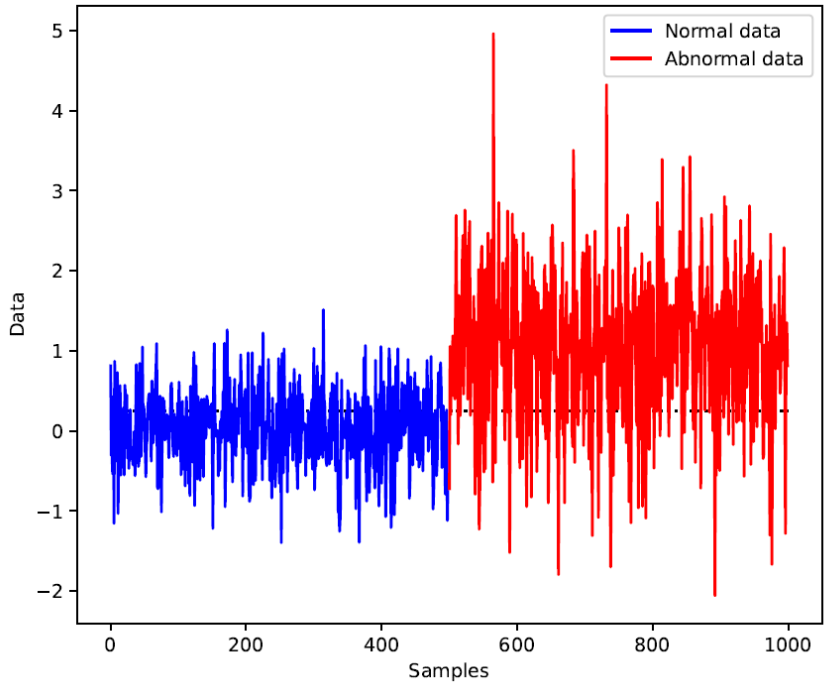

Figure 2.3: Data in normal and abnormal conditions for the example in Section 2.5.

the moving average filter and the median filter, denoted as $y_{MA}[k]$ and $y_{MED}[k]$ respectively, are given as follows:

$$y_{MA}[k] = \frac{1}{N} \sum_{m=k-N+1}^{k} x[m],$$
$$y_{MED}[k] = \text{Median}(\mathbf{x}[k]).$$

Here, we provide an example to compare the performance of different filtering methods. First, consider the data in normal and abnormal conditions with PDFs of $x_n \sim \mathcal{N}(0, 0.5)$ and $x_{ab} \sim \mathcal{N}(1, 1)$ respectively. Fig. 2.3 shows the data generated with these distributions.

Fig. 2.4 compares the PDFs of filtered data using different filtering methods. The PDFs of the filter outputs of the moving average filter, the median filter and the LLR filter for time-windows with the length of 10 data points are estimated using kernel density estimation. As it can be observed, the PDFs of normal and abnormal conditions using the LLR filter have the least area

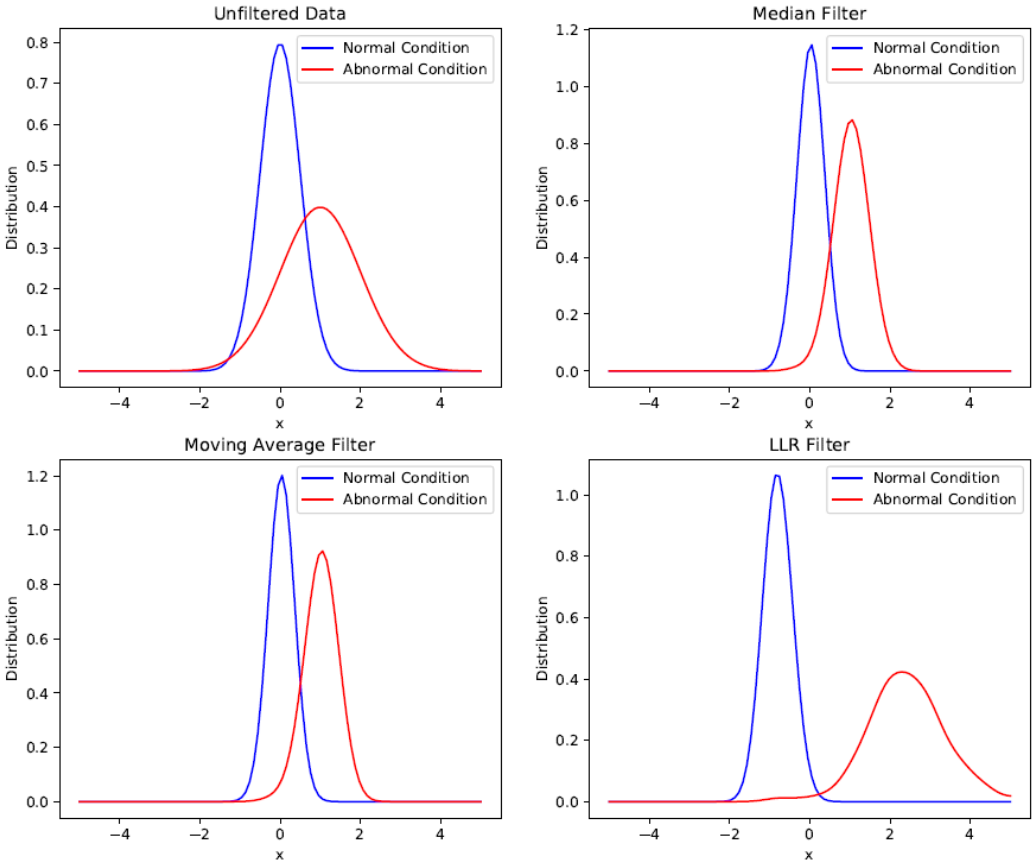

Figure 2.4: Comparison of estimated PDFs of the outputs of the filtered data in normal and abnormal conditions. The PDFs of filtered data are estimated using kernel density estimation.

of intersection. Therefore, the FAR and MAR of the LLR filter are the least among the different methods.

Chapter 3

A Two-Stage Deep-Learning Based Detection Method for Pipeline Leakage and Transient Conditions

In this chapter, a novel two-stage deep-learning based method is proposed to distinguish normal, leakage and transient conditions in pipelines. First, the problem of classification using the features of pressure and flow rate signals in a time-window is discussed. Further, the algorithm, consisting of preprocessing, stage 1 detection and stage 2 detection, is explained in detail. Finally, in the performance validation section, the datasets for training and testing the algorithm, the process of training the model and hyperparameter tuning and the test results are discussed.

3.1 System Description and Problem Statement

In this chapter, datasets of leak experiments from Suncor Energy Logistics Corporation are used to train and test the proposed algorithm. The data were obtained through online live monitoring from the Supervisory Control and Data Acquisition (SCADA) system. In these datasets, flow rates for the inlet

and the outlet are provided, as well as the pressure values at five measurement nodes including the inlet, the outlet and three middle nodes. In the proposed method, the flow difference signal between the inlet and the outlet sensors and the pressure signal from the inlet sensor are used.

In order to apply the machine learning based algorithm, time-series data are transformed into feature vectors over time-windows. A time-window length, denoted as L, is pre-set to split time interval into time-windows as follows:

$$TW(i) = [(i-1)s + 1,\ (i-1)s + L],$$

where $TW(i)$ is the vector containing indexes of data samples of the i^{th} time-window and s is the number of samples by which the time indexes change from one window to the next. The leak detection goal is to provide a label for each time-window using the feature vector extracted from data of each time-window. The feature vector for the ith time-window is denoted as $X(TW(i))$. This can be modeled as the following detection function:

$$y(i) = \Psi(X(TW(i))),$$

where $y(i)$ is the predicted label of $TW(i)$. There are three important conditions in a typical oil and gas pipeline: normal (steady-state), transient and leakage. These are the three possible labels for the leak detection output $y(i)$, which are also represented as 0, 1 and 2.

3.2 Proposed Algorithm

In this section, a two-stage data-based method is introduced to detect leak and transient conditions from pressure and flow rate difference signals. The diagram of the detection scheme is shown in Fig. 3.1, which has three main parts: preprocessing, Stage 1 detection, and Stage 2 detection. During preprocessing, time-series data is divided into time-windows. Then, feature vectors

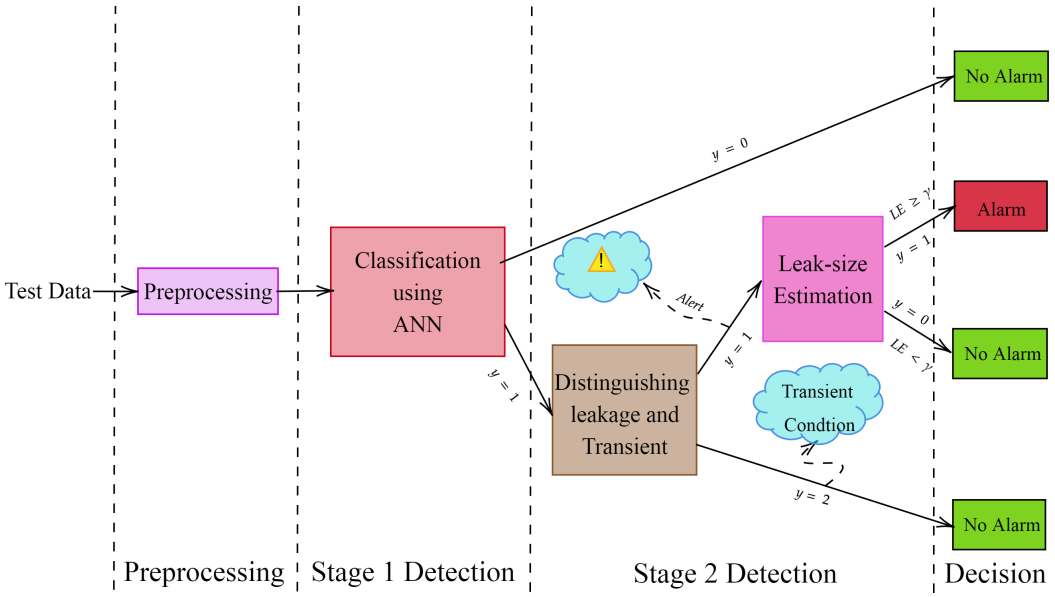

Figure 3.1: The proposed two-stage detection algorithm.

are extracted from the data of these time-windows and are normalized. In the first detection stage, an ANN is used to differentiate the normal class ($y = 0$) from the abnormal class ($y = 1$), including both the leak events and transient conditions. In the second detection stage, we distinguish leak and transient conditions via a simple logic between the flow rate difference and pressure change and apply a leak-size restriction. This stage helps to remove false alarms caused by transient and noisy time-windows, which is one major problem in current pipeline leakage detection methods. In the following subsections, we explain the three components in detail.

3.2.1 Data Preprocessing

The flow difference and pressure data are preprocessed through several steps before the detection stages. First, a sliding time-window with size L and moving lengths s are specified, and the data is grouped into a sequence of time-windows. Then, for each time-window, a feature vector is obtained. Afterward, in order to have a common scale for these features, we normalize the feature

vectors.

Feature Vector

As discussed in Section 3.1, a pressure drop and an increase in flow rate difference between the inlet and the outlet sensors are the key signatures of a typical leak in pipelines. Thus, the following two important features are used in this work.

- Mean of volume flow rate difference (denoted as X_1): For the k^{th} window, the mean of the volume flow difference between inlet and outlet sensors can be obtained as below:

$$X_1(k) = \frac{1}{L}\sum_{j=1}^{L}(\dot{V}_i(k_j) - \dot{V}_o(k_j)),$$

  where k_j is the jth sample of the kth time-window and L is length of the time-window.

- Mean of pressure difference (denoted as X_2): The mean of the pressure difference for the kth time-window is obtained as follows:

$$X_2(k) = \frac{1}{L}(P(k_L) - P(k_1)),$$

  where $P(k_i)$ is the ith measured pressure for the kth window at the inlet sensor.

Modified Hyperbolic-Tangent Estimator

Modified tanh estimator is one of the most efficient and robust approaches to perform normalization for neural networks. The main advantage of this method is its robustness to outliers. Denote the normalized feature vectors as $X_i^{(n)}$ for $i = 1, 2$. This estimator yields normalized data with the following expression:

$$X_i^{(n)}(k) = \frac{1}{2}\left(\tanh\frac{0.01\left(X_i(k)-\mu_i\right)}{\sigma_i}+1\right),$$

where $X_i^{(n)}(k)$ is the normalized value of $X_i(k)$. The parameters μ_i and σ_i which are the mean and standard deviation of the scores of the variable respectively [41]. In this method, we compute the mean and standard deviation directly through steady-state condition samples and use them in the normalization, which leads to good robustness.

3.2.2 Stage 1 Detection

The first stage of detection is to differentiate between normal and abnormal (leak or transient) conditions. In this stage, an ANN is trained using preprocessed training data and labels to perform classification. An ANN consists of an input layer, several hidden layers and an output layer, where each layer includes several neurons which are connected to the next layers by weights [42]–[44]. The weights are adjusted during the training process to learn the pattern between inputs and labels. The ANN input is composed of normalized feature vectors, i.e.,

$$X^{(n)} = [X_1^{(n)} \quad X_2^{(n)}],$$

where $X_1^{(n)}$ is the normalized vector of the mean of flow difference and $X_2^{(n)}$ is the normalized vector of the mean of pressure difference. For the output, the normal (steady-state) condition is labeled as '0', while both transient and leakage conditions are considered as an abnormality and labeled as '1'.

In our design, the hidden layers use the 'relu' activation function and the last layer uses the 'sigmoid' function, in order to predict the probability of abnormality in different time-windows. Thus, the output yields to the probability of abnormality $\hat{y}_p(k)$ in the k^{th} time-window. In other words,

$$\hat{y}_p(k) = \frac{1}{1+e^{-z(k)}},$$

where $z(k)$ is the input of the output layer for the k^{th} time-window obtained from a nonlinear relationship in the network as follows:

$$z(k) = \Xi(W, X^{(n)}(k)),$$

where $\Xi(.)$ is a nonlinear function of the hidden layers of the network, W is the vector containing network weights and $X^{(n)}(k)$ is the normalized feature vector of the k^{th} time-window in the testing dataset.

In order to perform classification, a threshold function is used to label each time-window as follows:

$$y_p(k) = \begin{cases} 0 & \hat{y}_p(k) < \alpha, \\ 1 & \hat{y}_p(k) \geq \alpha, \end{cases}$$

where $y_p(k)$ is the predicted label and α is a pre-defined threshold. The classification result varies on different values of α.

3.2.3 Stage 2 Detection

In the second stage of the detection, we separate transient and leakage conditions. In addition, we reduce false alarms using a pre-set leak-size tolerance. Here, we use two strategies to help remove false alarms caused by operational changes (transient conditions) and noisy data.

Distinguishing Transient Conditions Using Signs of Flow Change and Pressure Change

From our discussion in Section 3.1, an increase in the volume flow rate difference and pressure drop are the major characteristics of leakage conditions. However, after the initial pressure drop, the pressure stabilizes and the variation in pressure becomes similar to the steady-state condition. On the contrary, the pressure always varies during transient conditions such as pipeline shutdown or flow rate step-up processes. Therefore, the following logic-based

on the signs of the flow change and pressure change is used to distinguish leak and transient conditions.

Algorithm 1 Stage 2 detection

```
for Time window k labeled as y(k) = 1 do
    if X1(k) < 0 then
        y(k) = 2 ;                Step-down
    else
        if X1(k) ≥ 0 and X2(k) > λ then
            y(k) = 2 ;            Step-up
        else
            Alert = 1 ;
            if LE(k) ≥ γ then
                y(k) = 1 ;        Leakage
            else
                y(k) = 0 ;        Normal
            end
        end
    end
end
```

For the kth time-window, when the average flow difference is negative (i.e., $X_1(k) < 0$), the abnormality is labeled as a step-down (transient) condition (i.e., $y(k) = 2$). In the case of a positive average flow difference, it can be either leak or transient (step-up). If the pressure difference (i.e., $X_2(k)$) exceeds a pre-set threshold $\lambda > 0$, the time-window is classified as transient conditions (i.e., $y(k) = 2$). However, if the pressure drops or stabilizes (i.e., $X_2(k) \leq \lambda$), the time-window is considered as a possible leak and an alert is raised. To tune the threshold λ, we can set it to three times the standard deviation of X_2 under normal conditions. Therefore, if the pressure increase exceeds three times more than normal conditions, it is considered as a process step-up.

Alarm Removal Based on Leak-Size Tolerance

When a time-window is considered as possible leakage, an alert is raised in this stage. As noises can cause fluctuation in signal values, to avoid excessive false alarms caused by noisy data, an estimation of the leak size is obtained to

help handle the effect of temporarily noisy data. The leak size estimation for the k^{th} time-window is obtained using the sum of the flow rate difference in (m^3/h) for a sequence of alert starting from the k_0^{th} time-window as follows:

$$LE(k) = \frac{d}{3600} \sum_{j=l_0}^{l_k} (\dot{V}_i(j) - \dot{V}_o(j)),$$

where l_0 is the starting index of the k_0^{th} time-window, l_k is the last index of the k^{th} time-window and d is the sampling rate. When the estimated leak-size of a time-window sequence exceeds a pre-specified tolerance γ, an alarm is raised; otherwise, the alerts are neglected.

The overall scheme for the second stage of detection is summarized in Algorithm 1.

3.3 Performance Validation

3.3.1 Industrial Data and Detection System Setting

In this section, the proposed detection algorithm is tested with real industrial data from Suncor Energy Logistics Corporation in the offline mode. The data for training the ANN contains a total of 30 hours of measurements in which several leak experiments and operational changes were carried out. The test data were collected from another leak experiment that lasted for 12 hours including several operational changes, a small leak event and a large leak event. For both training and test data, upstream pressure and flow rate difference of both ends were collected every 5 seconds i.e., the sampling rate is 12Hz.

In the SCADA dataset, there were missed measurements for some sample times. In our experiments, the missed measurements are filled by the values of previous measurements. After the filling, the training and test datasets contain 21600 and 8640 samples for both variables respectively. In this experiment, the time-window parameters are set as $L = 24$ and $s = 6$. For the normalization,

Table 3.1: Comparison of different machine learning tools for Stage 1 detection.

Machine Learning Methods	FAR (%)	DR (%)	F1
Linear Logistic Regression	0.08%	90.06%	0.9444
DT	0.47%	92.71%	0.9428
SVM	0.23%	91.39%	0.9452
ANN	0.16%	91.39%	0.9485

the first 200 and 50 steady-state time-windows of training and test datasets are respectively used to estimate the mean and standard deviation in the steady-state condition. For Stage 1 detection, a multi-layer ANN is used with 3 hidden layers and every hidden layer has 10 neurons. The weights are adjusted using the Adam optimization method [45], with the batch size set to 72. One-tenth of the training data are used for validation and hyperparameters estimation.

3.3.2 Test Results

In order to compare the performance of different machine learning tools for Stage 1 detection, in Table 3.1, the FAR, DR and F1 score are shown for the ANN and three popular machine learning methods, where their hyperparameters are tuned with k-fold cross-validation using 10 folds. The leak-size tolerance is set to zero for all methods and α is 0.5 where it applies. As it can be observed, ANN has the best overall performance. The linear logistic regression has less false alarm rate; nonetheless, its DR is lower than other methods. The ANN classifier also has the highest F1 score.

Table 3.2 shows the sensitivity of the proposed method on the leak-size tolerance. As the leak-size tolerance is larger, the algorithm becomes more conservative in detection. Therefore, it leads to less false alarms, while the detection rate also decreases. As it can be seen in Table 3.2, when $\gamma = 0$, the algorithm reaches the highest DR and FAR. By increasing γ gradually, the false alarm rate decreases and no false alarm is received when $\gamma = 0.3$, with a reasonably high detection rate of 86.75%. The F1 score decreases as γ is set

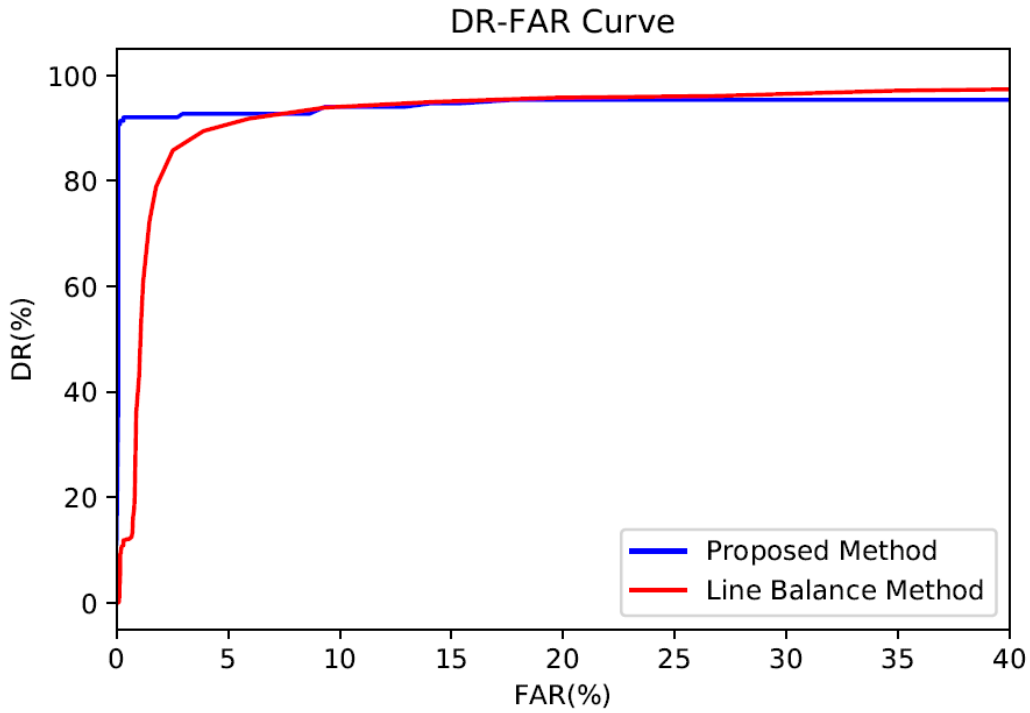

Figure 3.2: DR-FAR curves of the proposed method and the line balance method.

Table 3.2: Sensitivity of the proposed method to leak-size tolerance.

$\gamma(m^3)$	FAR (%)	DR (%)	F1
0	0.16%	91.39%	0.9484
0.1	0.16%	90.73%	0.9448
0.2	0.08%	88.74%	0.9371
0.3	0%	86.75%	0.9291

higher.

Further, Fig. 3.2 depicts the DR-FAR curves of the proposed method with $\gamma = 0$ and the line balance method. As it can be observed from Fig. 3.2, in the low FAR range, the DR of the proposed method is significantly higher than the line balanced method. For the case of FAR $\geq$ 10%, the DR of the line balance method is slightly higher, but the DRs of both methods are very close to 100%.

Fig. 3.3 and 3.4 illustrate the comparison of the performance between the

proposed method and the Kantorovich distance method [20] for small and large leakage scenarios respectively. In this comparison, both methods are using the inlet pressure and flow rate difference data. For the proposed method, α is set to 0.5. For the Kantorovich distance method, the threshold for flow difference residuals is set to be three times larger than the standard deviation of the first 500 steady-state flow rate difference data. The figures show the DD and FAR for varying leak-size tolerance values for the proposed method and varying Kantorovich distance thresholds for the inlet pressure and flow rate difference. As it can be observed in the case of the large leakage scenario (shown in Fig. 3.4), the proposed method results in less FAR for all detection delay values. However, for the small leakage scenario (shown in Fig. 3.3), the Kantorovich distance method has a smaller detection delay when FAR $\approx 0.08\%$. Besides, the proposed method is able to remove all false alarms (FAR = 0%) using a high leak-size tolerance and maintain the detection delays of 1 and 10 for large and small leakage scenarios respectively. However, the Kantorovich distance method cannot perform detection with no false alarms.

3.4 Summary

In this chapter, a two-stage leak detection algorithm using leak-signature based feature vectors and deep learning classification was proposed. The algorithm was implemented on an industrial dataset in an offline mode and the efficiency of the method was compared to two existing methods. The result showed that the proposed algorithm with ANN had the highest F1 score in comparison with the three machine learning methods. Further, by increasing the leak-size tolerance, the algorithm gave no false alarms while having 86.75% DR. Finally, the method was shown to have better overall performance in comparison with two existing methods, the line balance and Kantorovich distance methods.

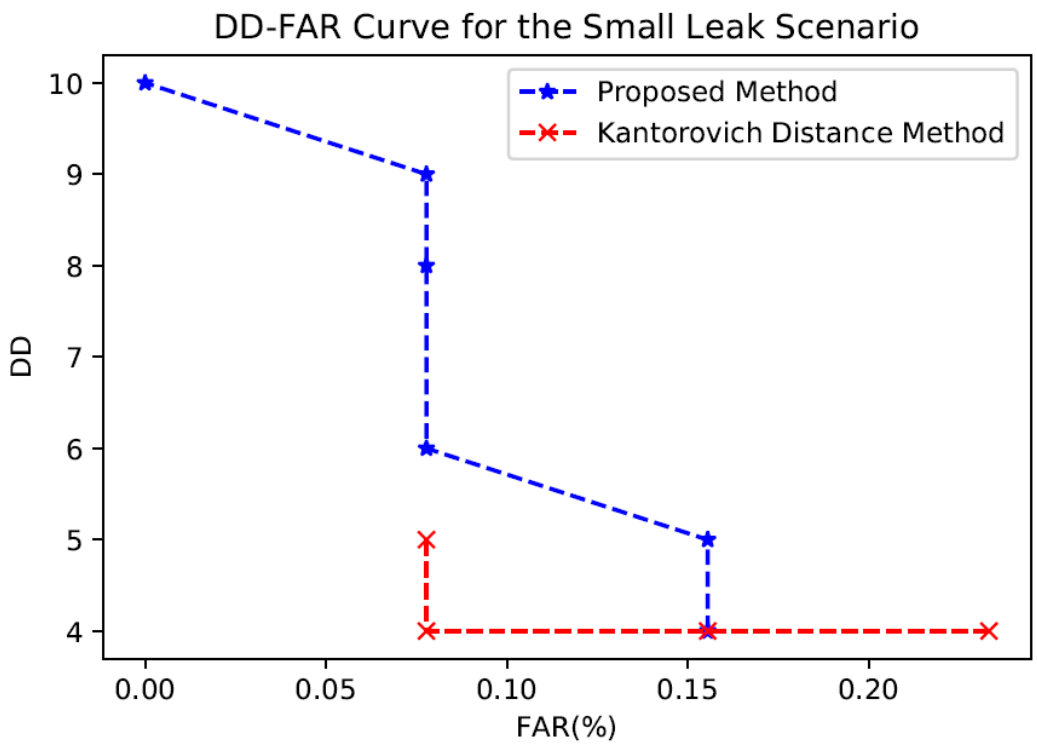

Figure 3.3: Detection delay of the small leak scenario versus FAR.

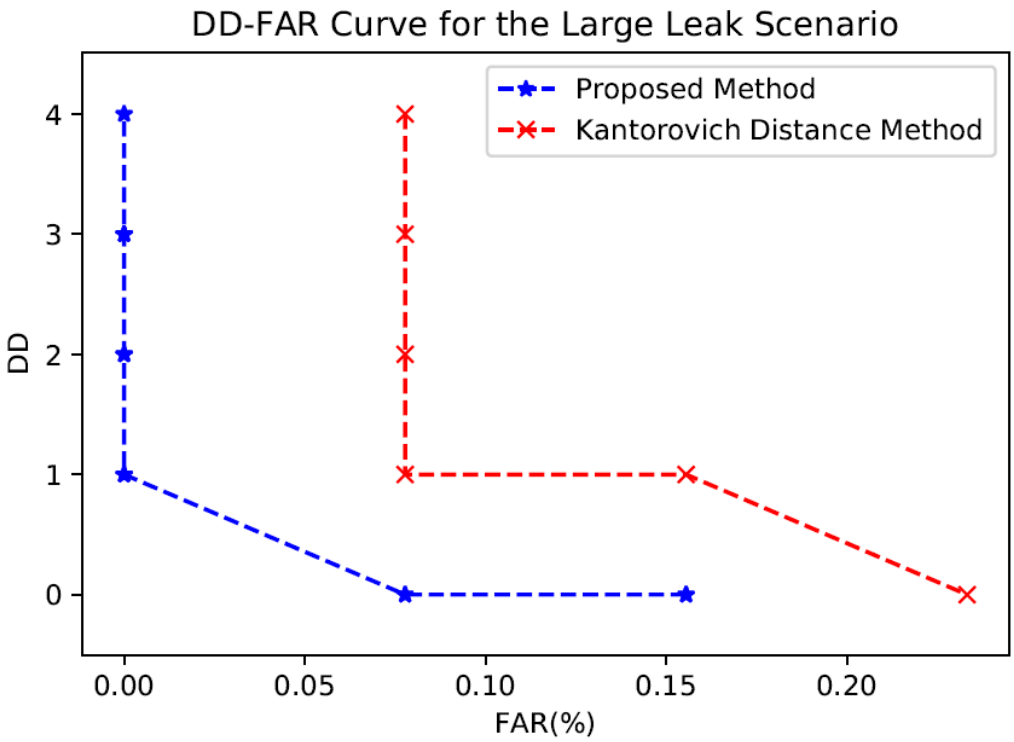

Figure 3.4: Detection delay of the large leak scenario versus FAR.

Chapter 4

Adaptive Naive Bayes Classifier Based Filter Using Kernel Density Estimation

In this chapter, an adaptive filter based on naive Bayes classifier and kernel density estimation is proposed to detect the changes in the mean value of the data in the positive direction, where the value of change is unknown. First, the detection problem of changes in the mean value of the data is formulated and the issue of employment of the optimal LLR filter is discussed. Then, the proposed method, including the initialization, pre-processing, filter design and density function update based on filter prediction, is explained. Also, the algorithm is customized for the application in the pipeline leakage problem. Further, the SNR in process faults is introduced and the necessity for density estimation or knowledge in implementing and tuning of different filters is discussed. Comparison with different benchmarks are conducted using simulated and real industrial data, and it is shown that the proposed filter has better overall performance in the detection of small leakage.

4.1 Problem Model and Assumptions

In this section, the problem of the detection of the change in the mean values of data in the positive direction is modeled. Let us assume that the PDF of

the data values, denoted as x, is shown as $f_n(x)$ for the normal condition. The goal is to trigger an alarm in the case of changes in the mean values of data in the positive direction, where the value of mean change is unknown. Therefore, there are two different scenarios for the data as follows:

- The data is in the steady-state condition or there is a change in the mean value of the data in the negative direction due to normal operation - no alarm should be raised for this scenario.
- There is a change in the mean values of the data in the positive direction due to faulty operation - an alarm should be raised for this situation.

As discussed in Section 2.1, as a leakage or process step-up occurs, the mean value of the flow difference between the inlet and the outlet sensors increases. In the case of the steady-state condition, there is no change in the mean value of the data, while in the process step-down condition, the mean value of data decreases. Therefore, the discussed scenario can differentiate between steady-state/step-down and leakage/step-up conditions. By applying a simple logic on the inlet pressure to differentiate leakage and step-up process, the problem can be associated with leakage detection in pipelines.

Here, we present these two scenarios in the form of two hypotheses on the data. For simplicity, we assume that the standard deviation of the data does not change in the faulty condition. Therefore, for the PDF $f_n(x)$, the PDF with positive and negative change in the mean value of the data is thus $f_n(x-\Delta)$. Denote $x[m]$ to be the sample value at time m, where $x[m]$ are independent and identically distributed (i.i.d.), the normal and faulty conditions are described as follows:

$$\mathcal{H}_0 : x[m] \; are \; i.i.d. \; f_n(x-\Delta), \; where \; \Delta \in (\Delta_{lo}, 0]$$

$$\mathcal{H}_1 : x[m] \; are \; i.i.d. \; f_n(x-\Delta), \; where \; \Delta \in (\Delta_{min}, \Delta_{up}).$$

The parameters $\Delta_{lo} < 0$, $\Delta_{min} > 0$ and $\Delta_{up} > 0$ are the maximum shift in the negative direction, the minimum shift in the positive direction and the maximum shift in the positive direction respectively. Further, define $C_{ab} = (\Delta_{min}, \Delta_{up})$ and $C_n = (\Delta_{lo}, 0]$, which are the sets of mean-shift values for the abnormal and normal classes, respectively. In this work, the value of Δ, the mean-shift, is modeled as a random variable with the PDF $f_\Delta(c)$. In addition, we consider to have an estimate of Δ_{min}, while Δ_{up} and Δ_{lo} are unknown.

The goal is to perform an accurate classification between these two scenarios, and to inform the operator in the case of detecting the faulty condition ($\mathcal{H}_1$) as soon as the fault transpires. Also, the operator should receive the least possible alarms in the normal operation.

4.2 Log-Likelihood Ratio Filter

Let us recall the LLR filter in Section 2.5. The LLR Filter solves this detection problem using the PDFs of the normal and abnormal conditions. Assuming that all normal samples follow $f_n(x)$ and abnormal samples follow the mean-shifted distribution of normal samples, $f_n(x-\Delta_1)$. The LLR filter is formulated as below:

$$y_o[k] = \sum_{m=k-N+1}^{k} \ln \frac{f_n(x[m] - \Delta_1)}{f_n(x[m])}. \tag{4.1}$$

where Δ_1 is the mean-shift value in the positive direction. The threshold can be set as $y_{tp} = \ln\left(\frac{c_1}{c_2}\right) = 0$ if $c_1 = c_2$. For the case that the data points follow a generalized Gaussian distribution in the normal condition, the optimal filter can be derived as follows. First, the PDF of the generalized Gaussian distribution is as below:

$$f(x) = \frac{\beta}{2\alpha\Gamma(1/\beta)} e^{-(|x-\mu|/\alpha)^\beta}$$

where μ is the mean of the PDF, $\alpha = \sqrt{\sigma^2 \frac{\Gamma(1/\beta)}{\Gamma(3/\beta)}}$, where σ^2 is the variance of the PDF and β is the shape parameter. By using the optimal filter rule and

incorporating the PDFs of normal and abnormal data, the following expression is driven for the LLR filter from Eq. (4.1):

$$y_o[k] = \sum_{m=k-N+1}^{k} ((|x[m] - \Delta_1 - \mu|/\alpha)^\beta - (|x[m] - \mu|/\alpha)^\beta).$$

For the case of Gaussian distributions (where $\beta = 2$), the filter can be further simplified as below:

$$y_o[k] = \frac{1}{\alpha^2} \sum_{m=k-N+1}^{k} (-2x[m]\Delta_1 - {\Delta_1}^2 - 2\mu\Delta_1).$$

As Δ_1 and μ are constant, by applying shift and scaling operations to the filter with respect to these variables, the optimal filter is equivalent to the moving average filter:

$$y_o[k] = \frac{1}{N} \sum_{m=k-N+1}^{k} x[m].$$

It can be seen that the optimal filter is independent of the mean-shift value (Δ_1). Nevertheless, the optimal threshold requires the knowledge of the mean-shift value as it undergoes the same shift and scaling operation with respect to Δ_1 and μ exerted on the filter to obtain the moving average filter.

Generally, the optimal filter cannot be simplified to the moving average filter for non-Gaussian distributions. In addition, the actual PDF is often unknown in a real industrial process. Hence, there are several issues in the implementation of the LLR filter:

- The exact PDFs of the normal and abnormal conditions may be unavailable.
- The PDF of the normal operation data may be changing during the process; for example, as a result of transient conditions.
- For the defined problem, the exact value of the change in mean value (Δ) is unknown, as it can be dependent on the size of the fault in pipeline leakage.

4.3 Proposed Fault Detection Method

In this section, a fault detection method based on naive Bayes classifier and kernel density estimation is proposed for the problem defined in Section 4.1. First, a collection of data is used to have a prior estimate of the CDF of the data in normal operation with $\Delta = 0$. Therefore, a naive Bayes based filter using the kernel density estimation is used over time-windows to perform the detection between $\mathcal{H}_0$ and $\mathcal{H}_1$ conditions. Then, predicted $\mathcal{H}_0$ samples using the threshold ($y_{tp} = 0$) are used as feedback values to update the primary collection. In order to avoid incorporating false negative predictions into the collection of data, a delay policy and a bounding constraint are also adopted. The method is named as "adaptive naive Bayes classifier based filter", abbreviated as the ANBC filter in the rest of the thesis.

4.3.1 Initialization

In this step, the number of sample points to estimate the CDF of steady-state condition, the minimum change in the mean value and the time-window length, respectively denoted as n_e, Δ_{min} and N, are pre-set. Afterward, the CDF of the steady-state condition is initialized. For this purpose, the approximate mean $\hat{\mu}$ and standard deviation $\hat{\sigma}$ of the data in the steady-state condition are required. As the underlying distribution is unknown, in the initialization stage, we assume that the data follows a Gaussian distribution. Thus, the initial estimate of kernel density estimation of steady-state condition is carried out using n_e Gaussian samples from $\mathcal{N}(\hat{\mu}, \hat{\sigma})$.

4.3.2 Preprocessing

Recall that $x[i]$ is the signal at the time instant i. The signal is segmented into time-window vectors with a length of N as below:

$$\mathbf{x}[k] = [x[k-N+1], x[k-N+2], ..., x[k]]\,,$$

where $\mathbf{x}[k]$ is the k^{th} time window. In industrial applications, some data values are missed in data measurement. Therefore, before segmentation, missing values are required to be filled. There are several approaches to fill the missing values such as the use of the previous values, interpolation, mean or median. The previous value method is more suitable for online implementation as the population mean, median and the next values in real-time signals may be unknown.

4.3.3 Filter and Threshold

In this section, the goal is to achieve a filter design to detect the mean value change in the positive direction (i.e., $\Delta > 0$) in the signal, while considering no change (i.e., $\Delta = 0$) and change in mean value in the negative direction (i.e., $\Delta < 0$) as normal condition. As there always exist noises and uncertainty in the data, a minimum detectable value (Δ_{min}) must be pre-determined based on system knowledge or the false alarm rate. Inspired by the naive Bayes classifier in Section 2.4, the probability of the C_n class, i.e., $\mathcal{H}_0$, given one signal value $x[m]$ is obtained as follows:

$$P(\mathcal{H}_0|x[m]) = \frac{\int_{C_n} f_n(x[m]-c) f_\Delta(c) dc}{\int_{C_n \cup C_{ab}} f_n(x[m]-c) f_\Delta(c) dc}.$$

Correspondingly, the probability of the C_{ab} class given the x is achieved as below:

$$P(\mathcal{H}_1|x[m]) = \frac{\int_{C_{ab}} f_n(x[m]-c) f_\Delta(c) dc}{\int_{C_n \cup C_{ab}} f_n(x[m]-c) f_\Delta(c) dc}.$$

Therefore,

$$\frac{P(\mathcal{H}_1|x[m])}{P(\mathcal{H}_0|x[m])} = \frac{\int_{\Delta_{min}}^{\Delta_{up}} f_n(x[m]-c)f_\Delta(c)dc}{\int_{\Delta_{lo}}^{0} f_n(x[m]-c)f_\Delta(c)dc}.$$

If we assume that the change in the mean value (Δ) follows a uniform distribution, the following equation is achieved:

$$\begin{aligned}\frac{P(\mathcal{H}_1|x[m])}{P(\mathcal{H}_0|x[m])} &= \frac{\int_{\Delta_{min}}^{\Delta_{up}} f_n(x[m]-c)dc}{\int_{\Delta_{lo}}^{0} f_n(x[m]-c)dc} \\ &= \frac{F_n(x[m]-\Delta_{min}) - F_n(x[m]-\Delta_{up})}{F_n(x[m]-\Delta_{lo}) - F_n(x[m])}.\end{aligned}$$

where $F_n(x)$ is the CDF with respect to $f_n(x)$. Further, assuming that Δ_{up} is very large and Δ_{lo} is very small, we have

$$\frac{P(\mathcal{H}_1|x[m])}{P(\mathcal{H}_0|x[m])} \approx \frac{F_n(x[m]-\Delta_{min})}{1-F_n(x[m])}.$$

With the consideration of the i.i.d. characteristic of the data, the following test statistic is proposed:

$$T(\mathbf{x}[k]) = \prod_{m=k-N+1}^{k} \frac{P(\mathcal{H}_1|x[m])}{P(\mathcal{H}_0|x[m])} \approx \prod_{m=k-N+1}^{k} \frac{F_X(x[m]-\Delta_{min})}{1-F_X(x[m])}.$$

In practical implementation, kernel estimation of the CDF $\hat{F}_X(x)$ is used. The proposed detection scheme becomes

$$y_o[k] = \tfrac{1}{N}\ln T(\mathbf{x}[k]) \approx \tfrac{1}{N}\sum_{m=k-N+1}^{k} \ln \tfrac{\hat{F}_X(x[m]-\Delta_{min})}{1-\hat{F}_X(x[m])} \gtrless y_{tp},$$

where y_{tp} is the threshold. Similar to the LLR filter, the best value of the threshold is $y_{tp} = 0$.

4.3.4 Density Function Update Based on Filter Prediction

In the initialization stage, since we had no assumption over the distribution of the data, the CDF of the steady-state condition with $\Delta = 0$ was estimated using a set of Gaussian samples. As the filter makes predictions on the data,

more data are revealed and the training set will be updated using the new data samples based on the prediction. In order to obtain a realistic approximation, the dataset for CDF estimation needs to be constantly updated with the most current data points in the steady-state. As the true labels of the data points are unknown, the filter prediction is used to update the dataset for CDF estimation.

A simple usage of all data that the filter predicts as $\mathcal{H}_0$ condition for the update has a few potential issues. First, the prediction errors, such as missed alarms, can bring inaccuracy in the dataset for CDF estimation. Second, when a fault occurs at the time m_l, based on the time-window segmentation, the most current sample of the time-window $\mathbf{x}[m_l]$ is the starting point of the fault condition; however, there are $N-1$ points in $\mathcal{H}_0$ condition in the time-window which can affect the filter detection and classify the time-window as $\mathcal{H}_0$ condition. Therefore, if $x[m_l]$ is included in the dataset for CDF estimation, it can cause errors in the CDF estimation. In addition, in the case of a change in the mean value of data in the negative direction, the filter detects the time-window as $\mathcal{H}_0$ condition. The inclusion of these samples in the CDF estimation can cause corruption. As a solution, we apply some constraints when including new samples for the CDF estimation. The update policy of the dataset for CDF estimation is explained as what follows.

Assuming that Z is the dataset for CDF estimation at time i shown as below:

$$Z[i] = [z_1, z_2, ..., z_{n_e}],$$

where $z_1, z_2, ..., z_{n_e}$ are the samples to estimate CDF using kernel density estimation. A delay D_u is pre-determined to update the set. In this algorithm, this value is tuned as $D_u = \left\lfloor \frac{N}{2} \right\rfloor + 1$, so that the D_u value is larger than half of the time-window length. Using the threshold $y_{tp} = 0$, if we have a sequence of all $\mathcal{H}_0$ predictions with length of D_u from window $i - D_u + 1$ to window i, we

select the data point $x[i - D_u + 1]$ as a candidate to be included in the dataset for CDF estimation. However, if the length of the sequence of $\mathcal{H}_0$ predictions is less than D_u, the dataset is not updated at the time instant i. In order to prevent outliers from entering into the dataset, a bounding constraint will be applied to the candidate points as below:

$$LB \leq x[i - D_u + 1] \leq UB$$

where LB and UB are respectively the lower bound and upper bound of the data in the steady-state condition. Tuning these parameters depends on the level of noise power in the data. Here, we applied three standard deviations of the current dataset samples from the mean of the training samples. If the data point is within the boundary region, it is included in the dataset; otherwise, it is not used to update $Z[i]$. When a new data sample is used to update the dataset, the least current point in the dataset is eliminated. Therefore, the updated dataset for CDF estimation, denoted as $\hat{Z}[i]$, is as follows:

$$\hat{Z}[i] = [z_2, ...z_{n_m}, x[i - D_u + 1]].$$

Finally, for the pipeline leak detection dataset in this chapter, we have an additional model-based constraint to prevent including corrupted data. In order to prevent including flow rate difference data points of the pipeline in the shut-down condition, the constraint $x[i - D_u + 1] \neq 0$ is validated before the update.

4.3.5 Online Leak Detection Scheme

In this section, an algorithm based on the pipeline flow rate and pressure signal is introduced in algorithm 2. In this algorithm, $y_o[k]$ is the output of the proposed filter on the k^{th} time window of the flow difference rate between the inlet and the outlet sensors. Therefore, $x[k]$ is the difference between inlet

and outlet flow rate at the kth time instant. In addition, $\Delta P_1[k]$ is defined as the pressure difference of the inlet sensor, obtaining as below:

$$\Delta P_1[k] = P[k] - P[k - N_p]$$

where $P[.]$ and N_p are the inlet pressure signal and the size of the inlet pressure signal time window respectively. As it was discussed in Section 2.1, the difference between flow rates of inlet and outlet sensors plays a pivotal role in leakage detection. After leakage transpires, the flow rate difference signal undergoes a positive mean shift fault, as depicted in Fig. 2.1. Therefore, the ANBC filter is used to detect the mean deviation. However, to exclude step-up conditions, there is a restricting condition on the positive pressure change in the inlet sensor, as in transient condition, the inlet sensor is affected earlier than the others. Therefore, using this method, step-up conditions do not raise leakage alarms. Figure 4.1 shows the integration of the ANBC filter and the leak detection method.

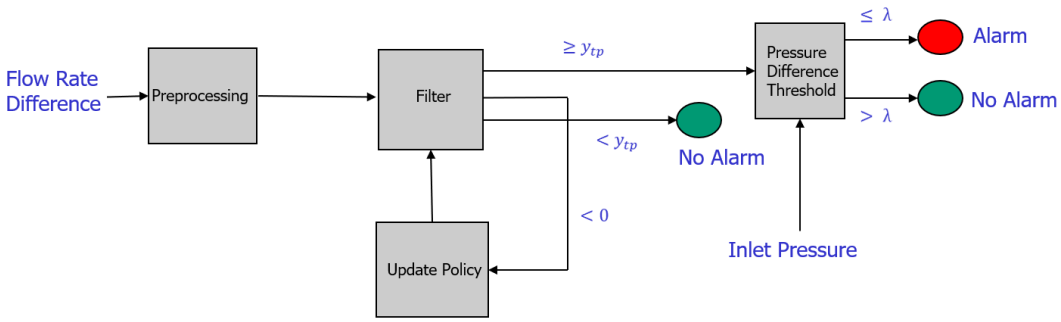

Figure 4.1: The schematic of the proposed leak detection scheme.

Algorithm 2 Leak detection algorithm

for *Time window k* **do**
  if $y_o[k] \geq y_{tp}$ *and* $\Delta P_1[k] \leq \lambda$ **then**
    Alarm
  else
    No Alarm
  end
end

4.4 Performance Measure

4.4.1 Signal to Noise Ratio in Process Fault

In this section, we introduce a measure for the fault intensity based on normal and abnormal conditions, the means and standard deviations. Intuitively, when the mean has a large change and the standard deviations of both the normal and/or abnormal conditions are lower, for any reasonable filter, the detection accuracy increases. On the contrary, a smaller change in the mean value and/or high standard deviations lead to low accuracy in the detection. Therefore, we introduce a new measure, inspired by the SNR concept in communication systems, as below:

$$\text{SNR} = 10 \log_{10} \frac{\mu_f^2}{\sigma_f^2}.$$

where $\mu_f = \mu_{ab} - \mu_n$ and $\sigma_f = \frac{\sigma_n + \sigma_{ab}}{2}$ in which μ_{ab}, σ_{ab}, μ_n and σ_n are the means and standard deviations of the normal and abnormal data respectively. Therefore, the performance of different filters can be compared for different fault SNRs.

4.4.2 Discussion on Implementation

In this chapter, the proposed method is compared to several benchmarks in the literature including LLR filters, moving average filters, median filters and unfiltered data. For the LLR filters, the exact PDFs of normal and abnormal conditions are required. Therefore, it is not feasible to implement them on industrial data. The moving average is one of the most industrial filters for detection. Median filters are also advantageous in detection, especially in the existence of outliers, which can affect the average of the samples. For the unfiltered data approach, there is no need to segment data into time-windows. The most current point is compared to a threshold. For moving average filters,

median filters and unfiltered data implementation, there is no necessity to have the exact or estimated PDFs. However, in order to obtain the optimal threshold, the exact or estimated PDFs of normal and abnormal conditions of the filtered data are required. Generally, the optimal threshold for these approaches is obtained as below [11]:

$$\frac{\partial J}{\partial y_{tp}} = -c_1 f_{Y_n}(y_{tp}) + c_2 f_{Y_{ab}}(y_{tp}) = 0$$

where J is the same loss function in Section 2.5. Therefore,

$$\frac{f_{Y_{ab}}(y_{tp})}{f_{Y_n}(y_{tp})} = \frac{c_1}{c_2},$$

with the same assumption of $c_1 = c_2$, the optimal threshold satisfies $f_{Y_{ab}}(y_{tp}) = f_{Y_n}(y_{tp})$. Therefore, for the analytical obtainment of the optimal threshold, the PDFs of the filtered normal and abnormal data are required. Overally, Table 4.1 shows the requirements of knowledge or estimation of density functions for implementation and optimal filter obtainment in different benchmarks.

Moreover, the exact CDF of the real time data is not mostly known. The ANBC filter estimates the real-time CDF of the steady-state condition in an adaptive fashion. However, it adds more complexity in the detection algorithm. The moving average filter, the median filter and unfiltered data are also more straightforward in terms of implementation. However, the optimal threshold is hard to estimate. In addition, the choice of the optimal filter and threshold depends on the distribution of the data. For example, the moving average and the median filters are shown to have a relatively desirable performance for the data following Gaussian and Laplace distributions respectively. Therefore, prior knowledge of the distribution is needed for the choice of these filters.

4.5 Simulation and Results

In this section, the proposed method is tested on both simulated and industrial pipeline leakage data. In the simulated data cases, a change in the mean value

Table 4.1: Necessity of density estimation or knowledge of normal and abnormal conditions for implementation and optimal threshold acquisition.

Method	Implementation	Optimal Threshold
ANBC Filter	✓	-
LLR Filter	✓	-
Moving Average	✗	✓
Median Filter	✗	✓
Unfiltered Data	✗	✓

in the positive direction is considered a fault scenario. However, in real industrial data, only leakage scenarios are taken into account and step-up processes are considered as normal conditions and they are excluded with exerting a preset threshold on the inlet pressure signal as shown in Section 4.3.5. In all the simulations, the transition between normal and abnormal conditions are taken into account. In other words, there are some time-windows containing both $\mathcal{H}_0$ and $\mathcal{H}_1$ conditions. In these time-windows, the true labels are obtained using the condition of the most current data point.

4.5.1 Simulated Data

In this section, we compare the performance of the proposed method with benchmarks in Table 4.1 in the literature in the sense of ROC curve and DR versus SNR for different distributions including Gaussian, uniform, Laplace and Gaussian mixture distributions. In all scenarios, the length of the time-window (N) is set to 10 and the update delay parameter (D_u) is tuned to 6. Also, for the ANBC filter, the initial values of the mean and standard deviation of the distribution are estimated using the first 50 steady-state samples. The Δ_{min} in the ANBC filter is set to 1. For the LLR filter used in this comparison, the real value of Δ is not given. The abnormal PDF for the LLR filter is obtained by shifting the normal PDF with a value of 1 as below:

$$p_{ab}(x) = p_n(x - \Delta_{LLR})$$

where the exact normal PDF is assumed to be known to the LLR filter and $\Delta_{LLR} = 1$.

ROC Curve

In this section, the ROC curves of different filters are shown to measure the performance in different distributions. In order to reduce the randomness in the results, the final curve for each method is obtained using its average in 1000 iterations. For ANBC filter, we used 80 data points to estimate the CDF ($n_e = 80$). In all cases, we have 160 data points in $\mathcal{H}_0$ condition, in which 20 data points are affected with a change in the mean value in the negative direction using samples generated by distribution $f_n(x - \Delta_{neg})$. The negative shift value (Δ_{neg}) is obtained using a uniform distribution of $\mathcal{U}\{-20, 0\}$. Also, we have 160 data in $\mathcal{H}_1$ condition with the distribution of $f_n(x-\Delta_{pos})$, in which the shift value (Δ_{pos}) is obtained using a uniform distribution of $\mathcal{U}\{0.8, 5\}$. Fig. 4.2 shows an example of the experimented scenario for the Gaussian distribution. The simulation is executed for 1000 times. In each iteration, the thresholds varied from the minimum filter output value to the maximum filter output value. The ROC curves are obtained using the average DR values and the average FAR values over all the iterations.

Fig. 4.3 shows the ROC curves of different methods with respect to the Gaussian distribution. For the PDF of the steady-state condition, the Gaussian distribution $\mathcal{N}(0, 2)$ is used. As it can be observed, the ANBC filter has better performance in comparison with the moving average, median filter and unfiltered data methods. The LLR filter has the best performance.

In the next scenario, for the PDF of the steady-state condition, a uniform distribution of $\mathcal{U}(0, 5)$ is used. As it can be observed in Fig. 4.4, the ANBC filter has better performance in comparison with all other methods, as its ROC curve is above other methods.

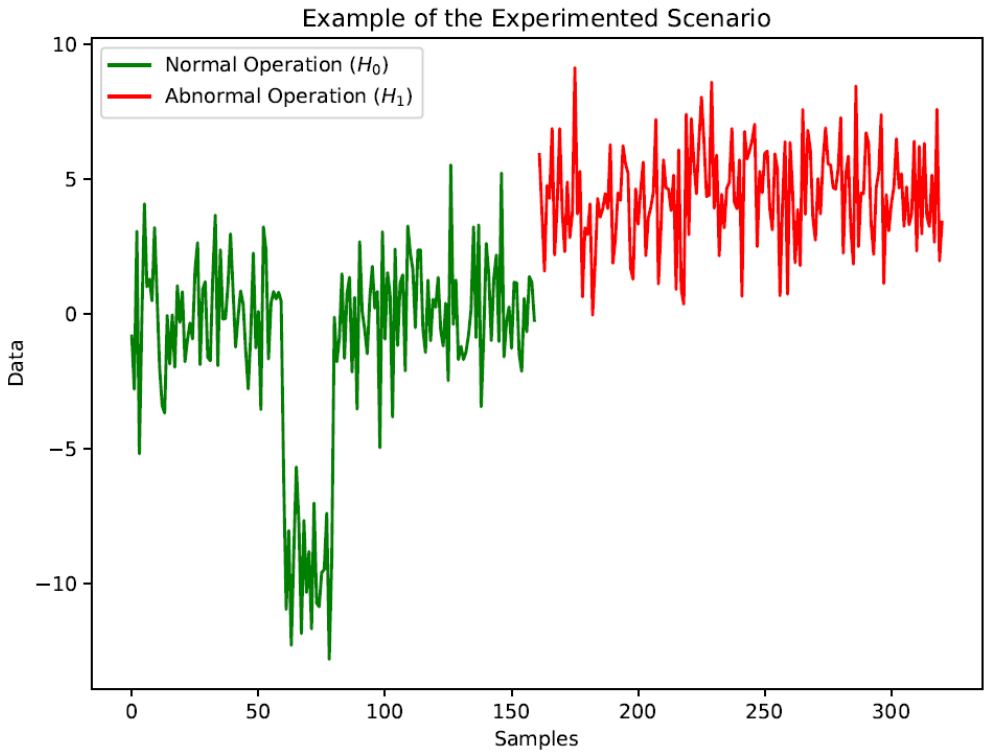

Figure 4.2: An example of the experimented scenario for Gaussian distribution.

In the third scenario, steady-state data are generated from the Laplace distribution $\mathbf{L}(0, 2)$. Fig. 4.5 shows that the performance of the ANBC filter is better than unfiltered data. The ROC curve of the ANBC method is very close to those of the moving average filter and median filter. Similarly, the LLR filter has the best performance over the compared methods.

Finally, a Gaussian mixture distribution with two components is used to generate the steady-state condition. The mean values of the components are set to 0.5 and -0.5. The standard deviation values of both components are 2. Fig. 4.6 shows the comparison between different methods. It shows that the ROC curve of the ANBC filter is above the moving average, median filter and unfiltered data. Similar to Gaussian and Laplace scenarios, the LLR filter has the best overall performance.

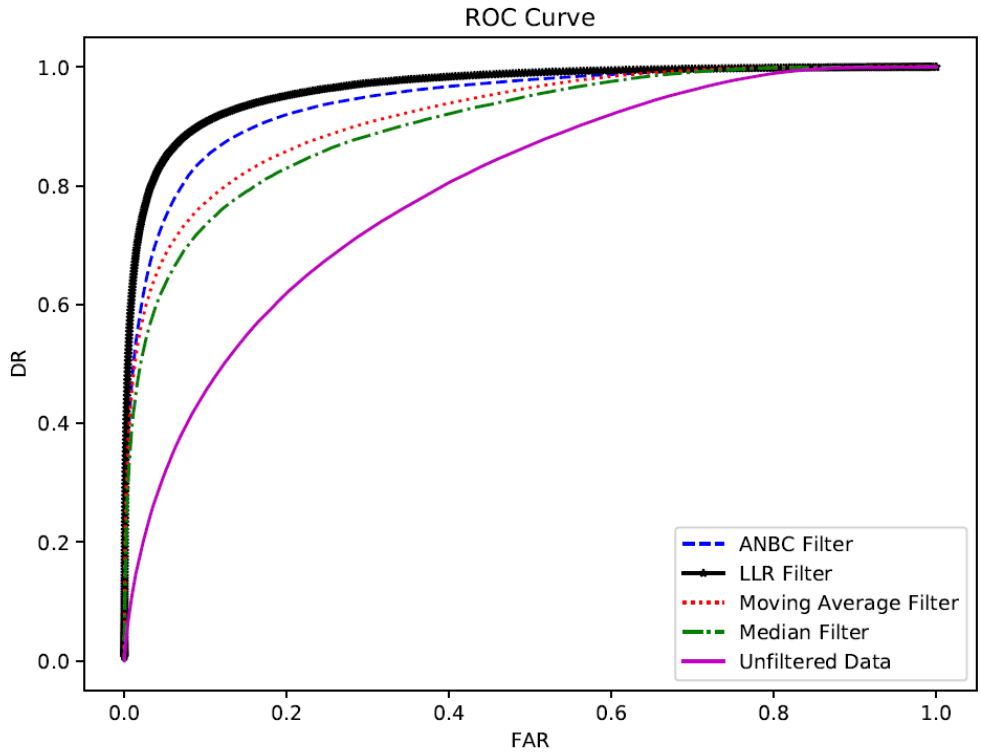

Figure 4.3: ROC curves of different methods for the case of Gaussian distribution.

DR versus different SNR Curve

In this section, the performance of different methods is evaluated using DR of different methods in FAR at 0.1 over different SNR values. In this experiment, n_e is set to 120 and we use 200 steady-state and 200 faulty data. The faulty data is generated by the shifted distribution of steady-state data with the shift value generated by a uniform distribution of $\mathcal{U}\{0.8, 1.2\}$. To generate different SNRs, we use different values for the standard deviation in each SNRs. For each standard deviation value, the simulation is iterated for 1000 times and the average DR and FAR values are obtained. Then, we obtain the DR at FAR value of 0.1 through interpolation of the average DR and FAR values. As the mean shift value is different in each iteration, the average of measured SNR

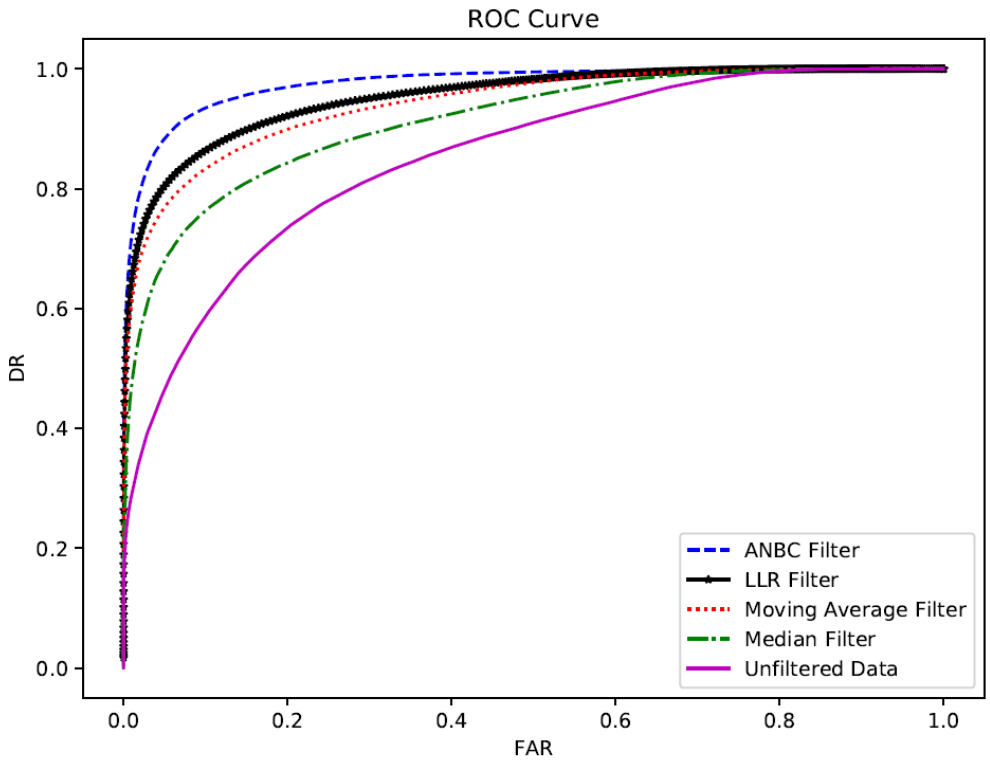

Figure 4.4: ROC curves of different methods for the case of uniform distribution.

values of all 1000 iteration is used to represent the SNR value corresponding to each standard deviation. Here, we perform this experiment for different distributions.

Fig. 4.7 shows the case of generating the steady-state data from the Gaussian distribution $\mathcal{N}(0, \sigma_G)$, where the value of σ_G varies for each SNR. The figure demonstrates that for low SNR values, the ANBC filter has higher DR at FAR value of 0.1 in comparison with other methods. For high SNR values, the LLR filter and moving average filter have almost the same performance as the ANBC filter.

The second scenario demonstrates the case of using the uniform distribution $\mathcal{U}(0, U_b)$ to generate the steady-state samples, where the value of U_b is adjusted to have desired standard deviation values. As it can be observed in Fig. 4.8,

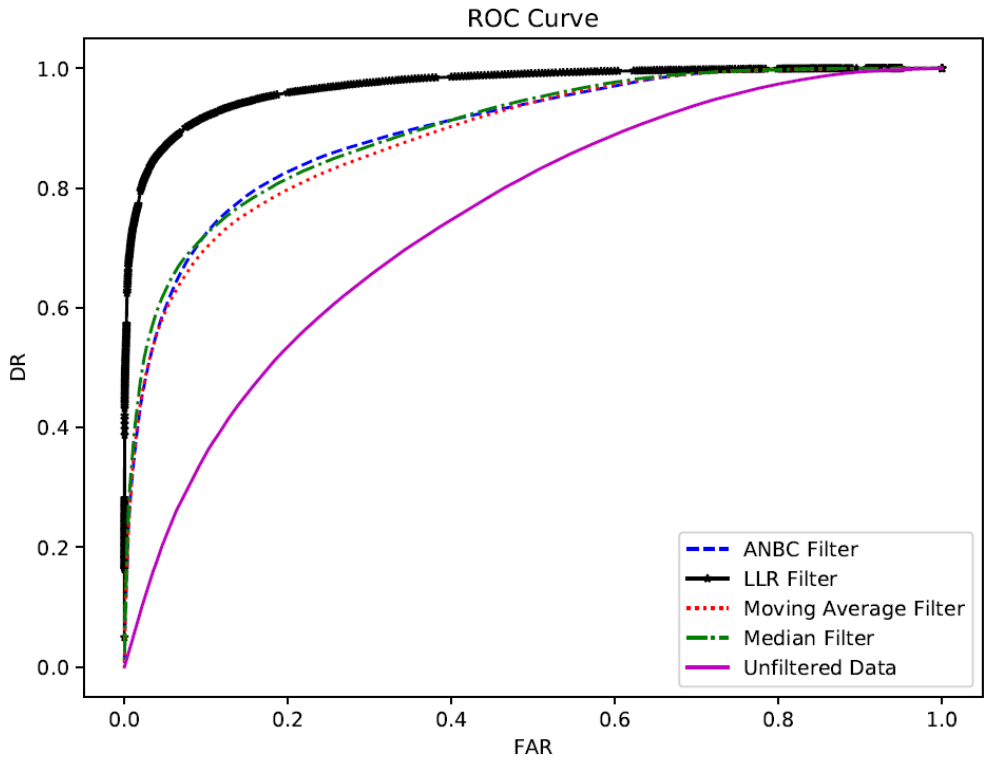

Figure 4.5: ROC curves of different methods for the case of Laplace distribution.

the ANBC filter has the best performance in the low SNR region. On the other hand, the DR of the LLR filter is higher for high SNR values.

In the next scenario, we use Laplace distribution $\mathbf{L}(0, S)$ to generate the steady-state data. The value of parameter S is adjusted to have the desired standard deviation values. Fig. 4.9 shows that the ANBC filter has the highest DR in comparison with other methods when the measured SNR is below -10dB. On the other hand, the median filter and LLR filter have better performance for higher SNRs.

Finally, the same comparison is made in the case of generating the steady-state data using a two-component Gaussian mixture distribution, where the means of the components are -0.5 and 0.5 and the standard deviations of the components are considered equal (σ_{GM}). The σ_{GM} is adjusted according to

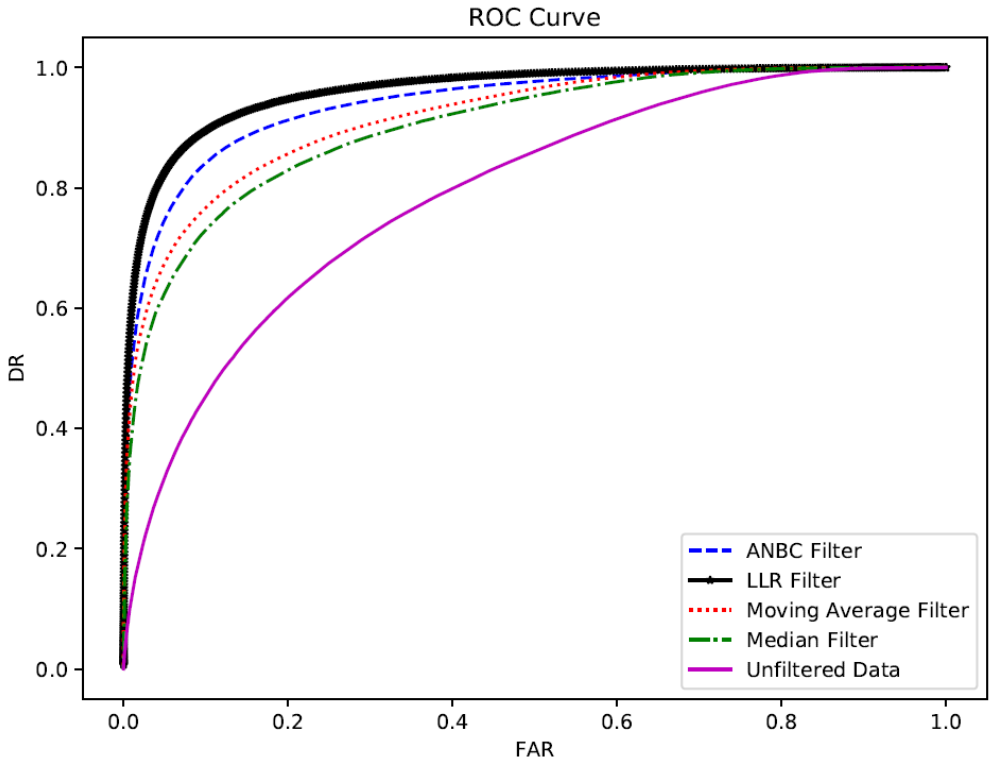

Figure 4.6: ROC curves of different methods for the case of Gaussian mixture distribution.

the desired SNR value. As Fig. 4.10 depicts, similar to the Gaussian case, in the low SNR region, the ANBC filter has the best DR at FAR value of 0.1 in comparison with other methods.

Table 4.2: Estimated SNR values of different leak scenarios.

Days	Day I		Day II	Day III	
Leak Scenarios	Leak I	Leak II	Leak I	Leak I	Leak II
SNR	0.43	-5.62	10.75	6.47	-1.79

4.5.2 Industrial Data

In this section, industrial data from Suncor Energy Logistics Corporation pipeline for three different days of leak experiments are used to evaluate the performance of the proposed filter with different benchmarks. The separation

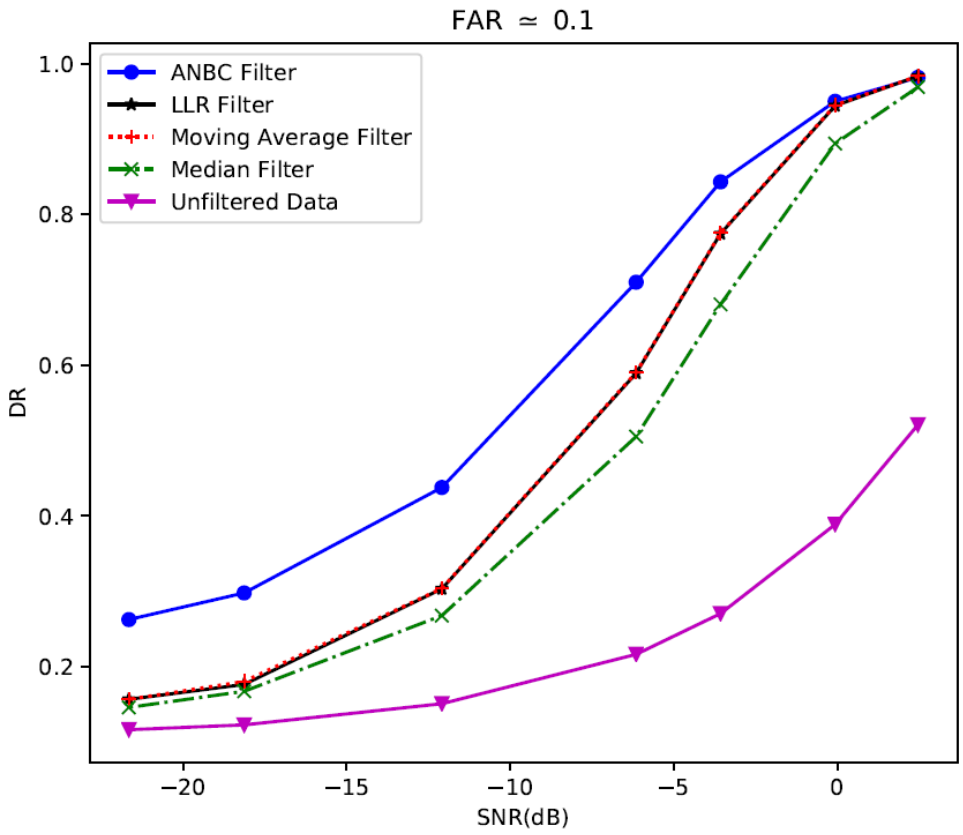

Figure 4.7: DR versus SNR for the case of Gaussian distribution at $FAR = 0.1$.

of the step-up and the leakage conditions are the same for all filters as shown in Algorithm 2. As the real distribution of the industrial data is not known, the LLR filter cannot be implemented here. In this experiment, the hyper-parameters are set to $L = 10$, $D_u = 6$, $n_e = 500$. In addition, Δ_{min} is tuned as 3.5 in all cases. Table 4.2 shows the leak scenarios in each day and their estimated SNRs.

Fig. 4.11, 4.12 and 4.13 show the ROC curves for Day I, Day II and Day III respectively. Fig. 4.11 and 4.13 show that the ROC curve of the proposed method is above other benchmarks for most of the FAR values, especially when the FAR is below 2%. By referring to Table 4.2, we can see that these two days' data contain leakages with lower SNR values. On the other hand, in Fig.

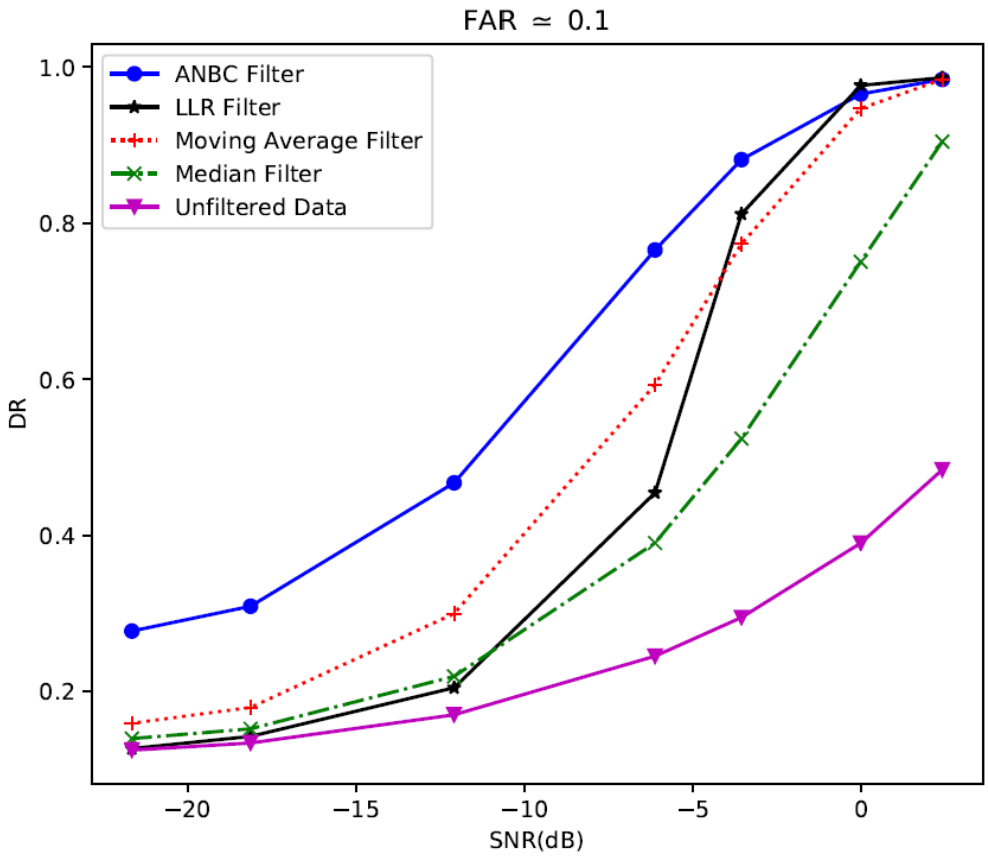

Figure 4.8: DR versus SNR for the case of uniform distribution at $FAR = 0.1$.

4.12, the performance of the moving average filter is marginally better than the ANBC filter for FARs below 2%.

Moreover, the detection delays of different methods are compared at FARs of 0.005, 0.01 and 0.02 for each leak scenario in Day I, Day II and Day III in Tables 4.3, 4.5, and 4.7 respectively. As it can be observed in Table 4.3, the DD values of the ANBC filter is less than other methods for the leak II in Day I, which means that the detection of small leakage is faster using the proposed method. Also, in Table 4.7, the DD values of the ANBC filter is less than other methods at FAR value of 0.005. For leak I in Tables 4.3, 4.5 and 4.7, the DDs of the unfiltered data method is less than other methods. The better performance of the unfiltered data method in the large leak (leak I) is

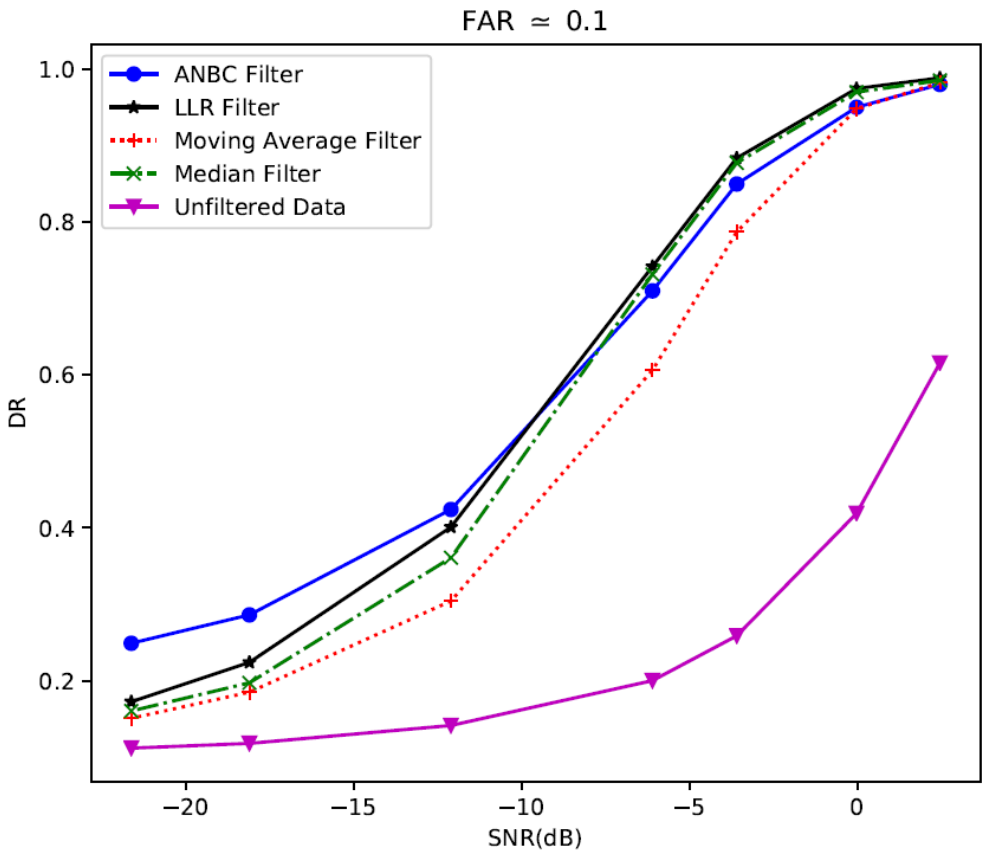

Figure 4.9: DR versus SNR for the case of Laplace distribution at $FAR = 0.1$.

because it only compares one sample point to the threshold, as opposed to other methods which perform detection based on time-windows.

Similarly, the DRs of different methods are compared at FARs of 0.005, 0.01 and 0.02 for each leak scenario in Day I, Day II and Day III in Tables 4.4, 4.6, and 4.8 respectively. As it can be observed in Tables 4.3 and 4.7, for leak II scenarios, the ANBC filter has the highest DRs in all three FAR values in comparison with other methods. Also, for the leakage scenario in Day II, the moving average filter has the highest detection rate at all three values of FAR, with the ANBC filter having equally the same detection rate in FAR values of 0.005 and 0.02. Similar to the DD comparison, the unfiltered data has the best overall performance in large leak scenarios in Tables 4.3 and 4.7.

Table 4.3: DD for each leakage and FAR of different methods in Day I data.

	DD of Leak I			DD of Leak II		
FAR	0.005	0.01	0.02	0.005	0.01	0.02
ANBC Filter	14	14	14	55	54	54
Moving Average Filter	13	12	12	81	69	68
Median Filter	14	14	14	80	68	68
Unfiltered Data	10	10	10	80	80	61

Table 4.4: DR for each leakage and FAR of different methods in Day I data.

	DR of Leak I			DR of Leak II		
FAR	0.005	0.01	0.02	0.005	0.01	0.02
ANBC Filter	0.7578	0.7578	0.7578	0.9410	0.9466	0.9469
Moving Average Filter	0.7623	0.7668	0.7668	0.9228	0.9349	0.9395
Median Filter	0.7578	0.7578	0.7578	0.9199	0.9315	0.9378
Unfiltered Data	0.7758	0.7758	0.7758	0.7515	0.8276	0.8739

Table 4.5: DD and FAR of different methods in Day II data.

	DD of Leak I		
FAR	0.005	0.01	0.02
ANBC Filter	37	35	34
Moving Average Filter	37	35	34
Median Filter	37	37	36
Unfiltered Data	33	33	33

Table 4.6: DR and FAR of different methods in Day II data.

	DR of Leak I		
FAR	0.005	0.01	0.02
ANBC Filter	0.9819	0.9828	0.9833
Moving Average Filter	0.9819	0.9832	0.9833
Median Filter	0.9794	0.9819	0.9824
Unfiltered Data	0.8593	0.8951	0.9272

Table 4.7: DD for each leakage and FAR of different methods in Day III data.

	DD of Leak I			DD of Leak II		
FAR	0.005	0.01	0.02	0.005	0.01	0.02
ANBC Filter	6	5	5	29	25	23
Moving Average Filter	4	4	3	30	26	23
Median Filter	4	4	4	29	26	25
Unfiltered Data	0	0	0	36	23	22

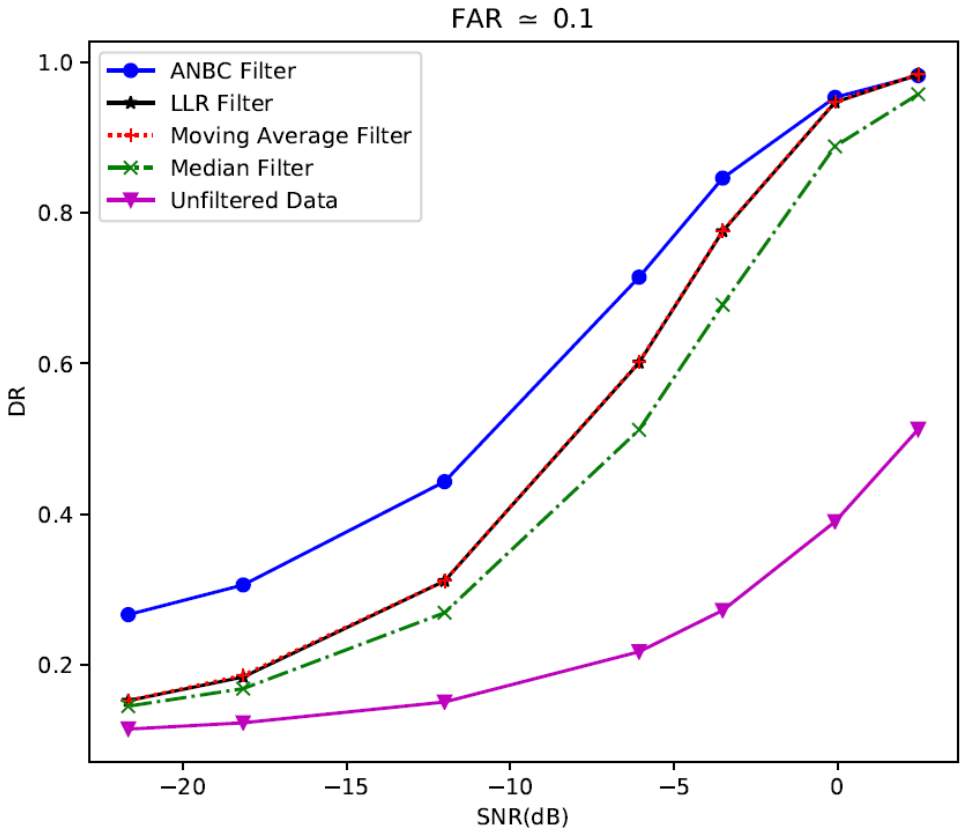

Figure 4.10: DR versus SNR for the case of Gaussian mixture distribution at $FAR = 0.1$.

4.6 Summary

In this chapter, a novel detection method was proposed for fault scenarios with a change in the mean value of data in the positive direction. The method uses kernel density estimation to approximate the CDF of the data in the steady-state condition with an adaptive approach. For the detection task, a naive Bayes based filter was used to detect the abnormalities. Also, the process for updating the data for estimating the CDF was proposed. The method was shown to have higher probability of detection values when $FAR = 0.1$ for four different distributions in relatively low SNRs. In addition, the ROC curves of the proposed method were compared to different benchmarks in the given scenarios. It was shown that the proposed method had better overall

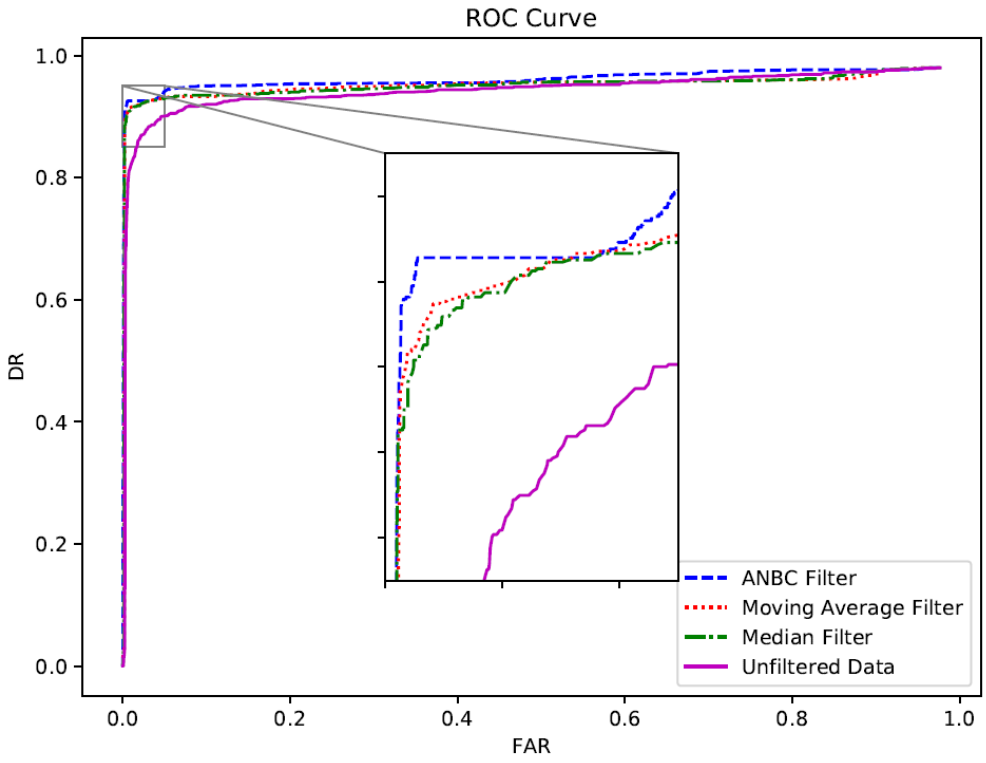

Figure 4.11: ROC curves of different methods in Day I data.

performance in comparison with the moving average filter, the median filter and the unfiltered data. Moreover, by using industrial data, it was shown that the proposed method had a better overall performance in detecting small leakages in terms of the detection rate and the detection delay at false alarm rate values of 0.005, 0.01 and 0.02. Overally, the proposed method was shown to have a good detection performance in the case of small changes in the mean

Table 4.8: DR for each leakage and FAR of different methods in Day III data.

	DR of Leak I			DR of Leak II		
FAR	0.005	0.01	0.02	0.005	0.01	0.02
ANBC Filter	0.7972	0.8042	0.8182	0.9364	0.9618	0.9649
Moving Average Filter	0.8112	0.8182	0.8322	0.8978	0.9604	0.9649
Median Filter	0.8112	0.8112	0.8112	0.8968	0.9512	0.9619
Unfiltered Data	0.8392	0.8392	0.8392	0.5665	0.7927	0.8825

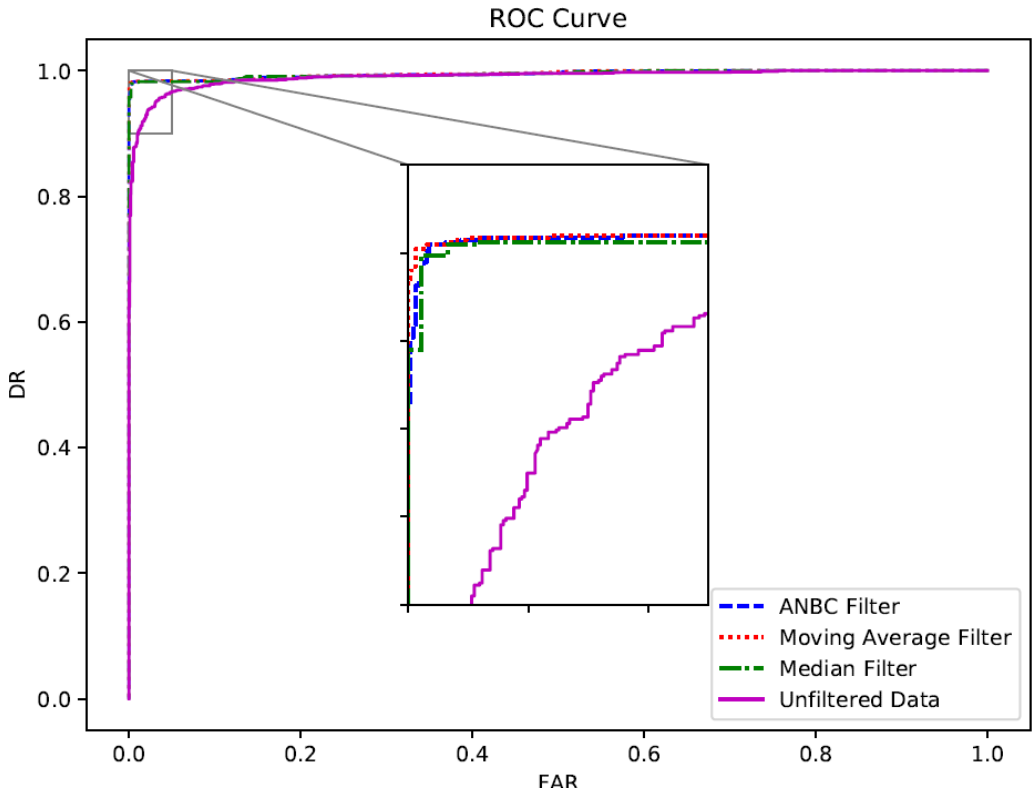

Figure 4.12: ROC curves of different methods in in Day II data.

value of the data in the positive direction.

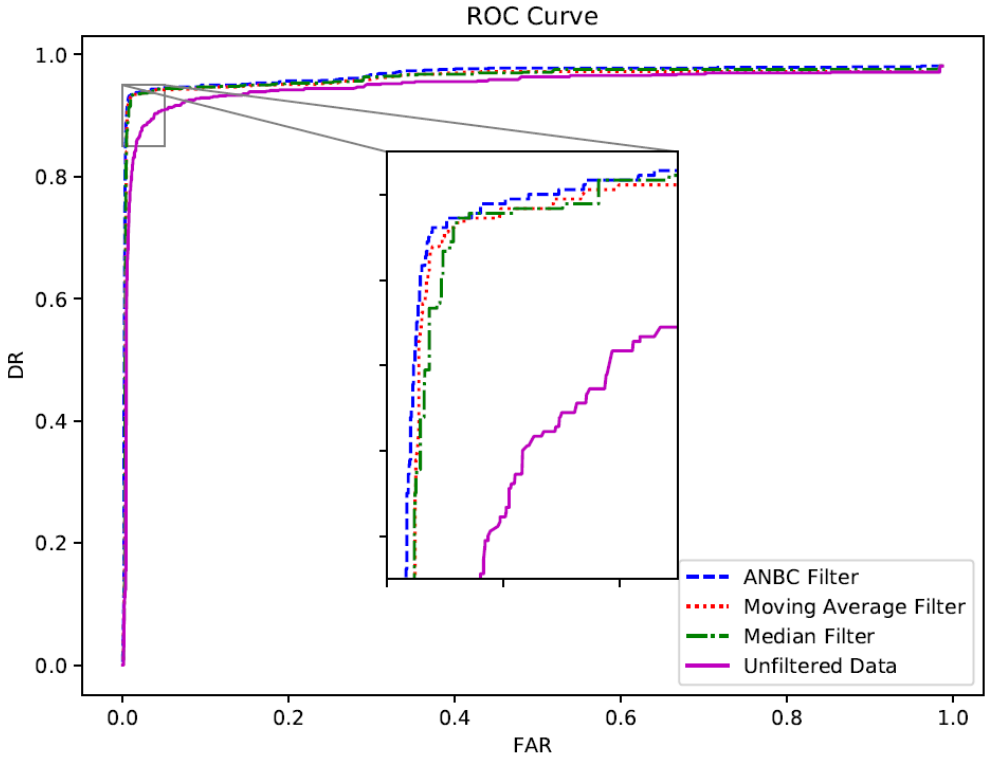

Figure 4.13: ROC curves of different methods in in Day III data.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this thesis, two methods were proposed to detect leakage in the oil pipelines. First, a two-stage deep learning based method was proposed to categorize normal, leakage and transient condition in the pipeline. The result was shown to have better overall performance in comparison to the two methods in the literature. As the leakage data is not always available to apply supervise learning based methods, an adaptive naive Bayes classifier filter using kernel density estimation was also proposed to detect changes in the mean value of the flow rate difference between inlet and outlet sensors data in the positive direction. In order to separate the step-up transient condition from the leakage, a simple logic was used on the inlet pressure difference signal. The proposed filter was applied on both simulated and industrial data and the results were compared to different benchmarks in the literature. It was shown that the proposed filter is more effective to detect small leakage scenarios in terms of the detection rate.

5.2 Future Work

Both proposed methods in this thesis can be improved further in future work. For the supervised learning method in Chapter 3, by using larger datasets with

more leakage scenarios and having more features such as acoustic signals, temperature, density, etc., the performance and the reliability can be improved. Also, more sophisticated deep learning models such as LSTMs can be trained and tested to increase the accuracy and robustness of the method.

In order to extend the proposed filter in Chapter 4, the assumption of independent distribution can be relaxed. Therefore, the conditional probabilities of data for the transition between the normal and abnormal conditions can be taken into account. Also, the proposed filter is designed to detect based on the model that the mean change is abrupt and consistent, although unknown. In reality, gradual changes exist, which can affect the robustness of the method by corrupting the dataset for CDF estimation. Therefore, both the filters and the update policy for the estimation of CDF can be improved to deal with gradual and abrupt changes. Finally, in order to improve the performance and robustness, an ensemble method using different filters can be considered.

References

[1] P. Goel, A. Datta, and M. S. Mannan, "Industrial alarm systems: Challenges and opportunities," *Journal of Loss Prevention in the Process Industries*, vol. 50, pp. 23–36, 2017.

[2] Y. Sun, W. Tan, and T. Chen, "A method to remove chattering alarms using median filters," *ISA Transactions*, vol. 73, pp. 201–207, 2018.

[3] I. Izadi, S. L. Shah, D. S. Shook, and T. Chen, "An introduction to alarm analysis and design," *IFAC Proceedings Volumes*, vol. 42, no. 8, pp. 645–650, 2009.

[4] I. Izadi, S. L. Shah, D. S. Shook, S. R. Kondaveeti, and T. Chen, "A framework for optimal design of alarm systems," *IFAC Proceedings Volumes*, vol. 42, no. 8, pp. 651–656, 2009.

[5] Y. Cheng, "Data-driven techniques on alarm system analysis and improvement," *PhD thesis, University of Alberta*, 2013.

[6] S. R. Kondaveeti, "Advanced analysis and redesign of industrial alarm systems," *PhD thesis, University of Alberta*, 2013.

[7] J. Xu, J. Wang, I. Izadi, and T. Chen, "Performance assessment and design for univariate alarm systems based on FAR, MAR, and AAD," *IEEE Transactions on Automation Science and Engineering*, vol. 9, no. 2, pp. 296–307, 2011.

[8] H. Xia, Z. Li, and X. Yang, "Optimal alarm threshold under time-varying operating conditions," in *2018 37th Chinese Control Conference (CCC)*, IEEE, 2018, pp. 5948–5953.

[9] N. A. Adnan, Y. Cheng, I. Izadi, and T. Chen, "A generalized delay-timer for alarm triggering," in *2012 American Control Conference (ACC)*, IEEE, 2012, pp. 6679–6684.

[10] E. Naghoosi, I. Izadi, and T. Chen, "A study on the relation between alarm deadbands and optimal alarm limits," in *Proceedings of the 2011 American Control Conference*, IEEE, 2011, pp. 3627–3632.

[11] I. I. Azad, "Alarm system design using rank order filters," *MSc thesis, University of Alberta*, 2015.

[12] M. Roohi and T. Chen, "Performance assessment and design of quadratic alarm filters," *IFAC-PapersOnLine*, 2020.

[13] Y. Cheng, I. Izadi, and T. Chen, "On optimal alarm filter design," in *2011 International Symposium on Advanced Control of Industrial Processes (ADCONIP)*, IEEE, 2011, pp. 139–145.

[14] Y. Cheng, I. Izadi, and T. Chen, "Optimal alarm signal processing: Filter design and performance analysis," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 2, pp. 446–451, 2013.

[15] L. M. Hall and J. P. French, "A modified cusum test to control postoutbreak false alarms," *Statistics in Medicine*, vol. 38, no. 11, pp. 2047–2058, 2019.

[16] W. Tan, Y. Sun, I. I. Azad, and T. Chen, "Design of univariate alarm systems via rank order filters," *Control Engineering Practice*, vol. 59, pp. 55–63, 2017.

[17] K. Rehman and F. Nawaz, "Remote pipeline monitoring using wireless sensor networks," in *2017 International Conference on Communication, Computing and Digital Systems (C-CODE)*, IEEE, 2017, pp. 32–37.

[18] P.-S. Murvay and I. Silea, "A survey on gas leak detection and localization techniques," *Journal of Loss Prevention in the Process Industries*, vol. 25, no. 6, pp. 966–973, 2012.

[19] T. Zhang, Y. Tan, X. Zhang, and J. Zhao, "A novel hybrid technique for leak detection and location in straight pipelines," *Journal of Loss Prevention in the Process Industries*, vol. 35, pp. 157–168, 2015.

[20] B. Arifin, Z. Li, S. L. Shah, G. A. Meyer, and A. Colin, "A novel data-driven leak detection and localization algorithm using the kantorovich distance," *Computers & Chemical Engineering*, vol. 108, pp. 300–313, 2018.

[21] O. Marceau and J.-M. Vanpeperstraete, "AUV optimal path for leak detection," in *OCEANS 2017-Anchorage*, IEEE, 2017, pp. 1–5.

[22] T. E. Barchyn, C. H. Hugenholtz, and T. A. Fox, "Plume detection modeling of a drone-based natural gas leak detection system," *Elem Sci Anth*, vol. 7, no. 1, 2019.

[23] M. A. Adegboye, W.-K. Fung, and A. Karnik, "Recent advances in pipeline monitoring and oil leakage detection technologies: Principles and approaches," *Sensors*, vol. 19, no. 11, p. 2548, 2019.

[24] L. Meng, Y. Li, W. Wang, and J. Fu, "Experimental study on leak detection and location for gas pipeline based on acoustic method," *Journal of Loss Prevention in the Process Industries*, vol. 25, no. 1, pp. 90–102, 2012.

[25] Q. Fu, H. Wan, and F. Qiu, "Pipeline leak detection based on fiber optic early-warning system," *Procedia Engineering*, vol. 7, pp. 88–93, 2010.

[26] G. M. Thompson and R. D. Golding, "Pipeline leak detection using volatile tracers," in *Leak Detection for Underground Storage Tanks*, ASTM International, 1993.

[27] G. Geiger, D. Vogt, and R. Tetzner, "State-of-the-art in leak detection and localization," *Oil Gas European Magazine*, vol. 32, no. 4, p. 193, 2006.

[28] A. bin Md Akib, N. bin Saad, and V. Asirvadam, "Pressure point analysis for early detection system," in *2011 IEEE 7th International Colloquium on Signal Processing and its Applications*, IEEE, 2011, pp. 103–107.

[29] M. B. Abdulla, R. O. Herzallah, and M. A. Hammad, "Pipeline leak detection using artificial neural network: Experimental study," in *2013 5th International Conference on Modelling, Identification and Control (ICMIC)*, IEEE, 2013, pp. 328–332.

[30] D. Zang, J. Liu, and H. Wang, "Markov chain-based feature extraction for anomaly detection in time series and its industrial application," in *2018 Chinese Control And Decision Conference (CCDC)*, IEEE, 2018, pp. 1059–1063.

[31] M. Zadkarami, M. Shahbazian, and K. Salahshoor, "Pipeline leakage detection and isolation: An integrated approach of statistical and wavelet feature extraction with multi-layer perceptron neural network (MLPNN)," *Journal of Loss Prevention in the Process Industries*, vol. 43, pp. 479–487, 2016.

[32] P. Xu, R. Du, and Z. Zhang, "Predicting pipeline leakage in petrochemical system through GAN and LSTM," *Knowledge-Based Systems*, vol. 175, pp. 50–61, 2019.

[33] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[34] A. C. Guidoum, "Kernel estimator and bandwidth selection for density and its derivatives," *Package "kedd"*, 2015.

[35] V. A. Epanechnikov, "Non-parametric estimation of a multivariate probability density," *Theory of Probability & Its Applications*, vol. 14, no. 1, pp. 153–158, 1969.

[36] M. C. Jones, J. S. Marron, and S. J. Sheather, "A brief survey of bandwidth selection for density estimation," *Journal of the American Statistical Association*, vol. 91, no. 433, pp. 401–407, 1996.

[37] A. Z. Zambom and R. Dias, "A review of kernel density estimation with applications to econometrics," *arXiv preprint arXiv:1212.2812*, 2012.

[38] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT press, 2012.

[39] I. Rish *et al.*, "An empirical study of the naive Bayes classifier," in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, 2001, pp. 41–46.

[40] D. Berrar, "Bayes' theorem and naive Bayes classifier," *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, vol. 403, 2018.

[41] L. Latha and S. Thangasamy, "Efficient approach to normalization of multimodal biometric scores," *International Journal of Computer Applications*, vol. 32, no. 10, pp. 57–64, 2011.

[42] B. Yegnanarayana, *Artificial Neural Networks*. PHI Learning Pvt. Ltd., 2009.

[43] M. H. Hassoun, *Fundamentals of Artificial Neural Networks*. MIT press, 1995.

[44] A. K. Jain, J. Mao, and K. M. Mohiuddin, "Artificial Neural Networks: A Tutorial," *Computer*, vol. 29, no. 3, pp. 31–44, 1996.

[45] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.