Czech Technical University in Prague

Faculty of Mechanical Engineering

Department of Instrumentation and Control Engineering

DOCTORAL THESIS

Novelty detection via linear adaptive filters

Ing. Matouš Cejnek

Doctoral programme: *Control and Systems Engineering*

Supervisor: *doc. Ing. Ivo Bukovský, Ph.D.*

Prague 2020

I hereby declare I have written this doctoral thesis independently and cited all the sources of information used in accordance with methodological instructions on ethical principles for writing an academic thesis. Moreover, I state that this thesis has neither been submitted nor accepted for any other academic degree.

Prohlašuji, že jsem tuto práci vypracoval samostatně a citoval jsem všechny použité informační zdroje v souladu s metodologickými a etickými principy platnými pro psaní akademické práce. Dále prohlašuji, že tato práce nebyla podána nebo přijata pro získání jakéhokoliv jiného akademického titulu.

In Prague, January 2020 Matouš Cejnek

Acknowledgment

The work presented in this thesis was supported by various grants. The particular grants can be found in original articles and proceedings cited in chapter *Author's references*.

Throughout the writing of this dissertation, I have received a great deal of support and assistance. I would first like to thank my supervisor, doc. Ing. Bukovský Ivo Ph.D., whose expertise was invaluable in the formulating of the research topic and methodology in particular. I would like to acknowledge my colleagues for their wonderful collaboration: Jan Vrba, Zdeněk Novák, Cyril Oswald, Adam Peichl and others.

In addition, I would like to thank my parents for their wise counsel. Finally, there are my friends, who were of great support in deliberating over our problems and findings, as well as providing happy distraction to rest my mind outside of my research.

Abstract

Novelty detection is an important signal processing task. This task is essential for many industry, and biomedical applications. This thesis is presenting research on the topic of novelty detection utilizing parameters of linear adaptive filters. A new method of adaptive novelty detection is presented in this thesis - Error and Learning Based Novelty Detection. The goal of this thesis is to present the new method as a viable tool for online unsupervised novelty detection in non-stationary and drifted data. The method is supported with various experimental evidence collected from multiple studies. These studies cover multiple traditional applications like system change point detection and outlier detection. The results are obtained from experiments with real and synthetic data.

Abstrakt česky

Detekce novosti je důležitá část zpracování signálů a je esenciální pro různé průmyslové a bioinženýrské aplikace. Tato disertace prezentuje výzkum metod detekce novosti využívající parametry adaptivních filtrů. V této práci je popsána nová methoda adaptivní detekce novosti nazvaná Error and Learning Based Novelty Detection. Cílem této práce je popsat tuto novou metodu jako užitečný nástroj pro online detekci novosti pro data, která jsou nestacionární nebo obsahují drift. Studie podporující užitečnost této metody jsou představeny v této práci. Tyto studie pokrývají různé oblasti tradičního zpracování signálů, například: detekce změny chování systému a detekce anomálií. Experimentální výsledky těchto studií jsou získány na reálných i syntetických datech.

Contents

Nomenclature

$\bar{x}(k)$	Mean value of input vector in discrete time k	
$\Delta\mathbf{w}(k)$	Vector of adaptive weights increments in discrete time k	
δ	Initialization parameter	[1]
ϵ	Regularization term (small positive constant (NLMS, NLMF)	[1]
η	Normalized learning rate (small positive constant)	[1]
γ	Forgetting factor	[1]
μ	Learning rate	[1]
ρ	Hyperparameter of GNGD	[1]
σ_x	Standard deviation of input values	[1]
σ_y	Standard deviation of target	[1]
$\mathbf{I}$	Identity matrix	
$\mathbf{nd}(k)$	Vector of novelty descriptors in discrete time k	
$\mathbf{R}(k)$	Auto-correlation matrix in discrete time k	
$\mathbf{w}(k)$	Vector of adaptive weights in discrete time k	
$\mathbf{X}$	Matrix of all input data	
$\mathbf{x}(k)$	Input vector in discrete time k	
$\mathbf{x}_c(k)$	Centered input vector in discrete time k	

$\tilde{y}(k)$	Adaptive model output	[1]
$\vec{1}$	Vector of all ones	
$e(k)$	Error of adaptive model in discrete time k	[1]
$h_i(k)$	i-th parameter of data generator in time k	
k	Discrete time index	[1]
$nd(k)$	Result of novelty descriptors reduction in discrete time k	[1]
o_i	i-th operation	
$v(k)$	Additive noise	
w_i	i-th adaptive weight in discrete time k	[1]
x_i	i-th adaptive model input in discrete time k	[1]
$y(k)$	Adaptive model target	[1]
$y_t(k)$	Z-scored target in discrete time k	[1]
ACC	Accuracy	
ANN	Artificial neural networks	
AP	Affine projection	
ApEn	ApEn	
AUROC	Area under the receiver operating characteristic	
AWGN	Additive white Gaussian noise	
ECG	Electrocardiography	
EEG	Electroencephalography	
ELBND	Error and Learning Based Novelty Detection	
FD	Fuzzy Density	

FN False negative

FP False positive

FPR False positive rate

GMM Gaussian mixture models

GNGD Generalized normalized gradient descent

HF High Frequency power in ECG: frequency activity in the 0.15 - 0.40Hz range

HMM Hidden Markov models

HONU Higher order neural units

IoT Internet of Things

IQR Interquartile range

KNN K-nearest neighbour algorithm

LE Learning entropy

LF Low Frequency power in ECG: frequency activity in the 0.04 - 0.15Hz range

LMF Least mean fourth

LMS Least mean squares

LNU Linear neural unit

MD Mahanobilis distance

MEG Magnetoencephalography

MLP Multi-layer perceptron

mV milivolt

NLMF Normalized least mean fourth

NLMS Normalized least mean squares

NSSLMS Normalized sign-sign least mean squares

OCNLMS Online centered normalized least mean squares

PCA Principal component analysis

PPV Precision

PQRST ECG wave: P wave followed by the QRS complex and the T wave

RLS Recursive least squares

RNN Replicator neural network

ROC Receiver operating characteristic

SE Sample Entropy

SEN Sensitivity

SNR Signal to noise ratio

SOM Self-organizing map

SPE Specificity

sps Samples per second

SSLMS Sign-sign least mean squares

SVDD Support vector data description

SVM Support vector machine

TN True negative

TP True positive

TPR True positive rate

List of Figures

List of Tables

Chapter 1

Introduction

1.1 What novelty detection is

Novelty detection is the name for identification of something new or unknown in data. The exact meaning of event described as something new depends on its application and on the field. However, in general the novel event is something that is not expected in the data because of the data generating process nature.

The task of novelty detection is one of the oldest and the most fundamental tasks in *machine learning* field. The monitoring of production process or of any other process is a costly work if it is done by a human operator. That is the reason why huge effort to automate this process has been done in the last few decades. Despite of this fact, the term novelty detection has started appearing in literature after year 2000. Although the novelty detection topic is that old, the conquest of novelty detection algorithms development has not yet finished. With every new technology or ability to measure, transfer and store data that mankind posses a new novelty detection challenges emerge.

Today, the term of novelty detection is used as a broader term for detection of various novel events - anomaly detection [1], outlier detection [2], fault detection [3], novel class detection [4], concept drift detection [5]. The need for multiple more specific names is caused by the fact, that the objective of novelty detection task can significantly differ among various novelty detection applications. In some cases

the goal of novelty detection can be perturbation detection (one-sample-outliers) in gradually changing environment. In other case, the goal may be detection of gradual changes of environment by itself while the perturbations are ignored. Various categorizations of novelty detection are often used. Probably the most fundamental categorization is based on the scale of the detection [6]. The categories are:

- contextual novelty detection - can be understand as system change point detection, or detection of change in process that is generating the data;
- value based novelty detection - this name stands for detection of various perturbations, or generally short-time events, that does not belong into expected behavior of the observed system.

1.2 Importance of novelty detection

Nowadays the novelty detection is crucial in many fields. The demand is even increasing in the last decade. This is caused by the increase in production of data streams in modern world [7]. This increase is related to modern concepts like *Internet of Things* (IoT) [1] and *Big Data* [2]. The data streams need to be often monitored online as fast (with low lag) as possible. This is required for example in medical diagnostic, process control and market fluctuations monitoring. In some scenarios the novelty detection is an important mechanism used as the safety stop mechanism for processes that are hard to control. Such mechanism can stop a process in case when it reaches out of the planned or safe boundaries.

In different scenario, a novelty detection mechanism can be used as a medical diagnostic tool for detection of malfunctioning organ symptoms in a patient. In other scenario, the novelty detection can adjust learning rate of a machine learning algorithm according to the level of novelty in data. In general it is possible to sum-

[1] IoT is the network of devices, vehicles, home appliances and other items embedded with electronics, software, sensors or actuators.

[2] Big Data is unofficial definition term commonly used to describe data sets that are so voluminous and complex that traditional data-processing application software are inadequate to deal with them.

marize that the novelty detection is an automated tool for tasks that cost significant amount of time, require high level of focus and or have only small error tolerance.

1.3 Adaptive novelty detection

Adaptive novelty detection is a special case of novelty detection. It is a special case because it is featuring adaptive or learning algorithms. In some broader sense, the learning algorithm function can be understood as a compression of information from data into adaptive parameters. Such a compressed information in form of a smaller number of parameters can be processed much faster than full scale data. Furthermore, the process of learning can also highlight the important features from data and reflect them in adaptive parameters or their increments. Another interesting feature of adaptive models is their prediction or classification error. Such an error can provide valuable information about novelty hidden in data. An adaptive novelty detection method can use error of the adaptive model, or increment of adaptive parameters, or both, to determine how novel the particular samples are. Summary of the key features of adaptive models that are desirable during novelty detection process is as follows:

- *Compression* - the adaptive algorithms has the ability to describe a long window of historic information in smaller number of parameters of their updates.
- *Prediction* - the ability to predict few samples ahead can be useful to minimize the delay between sample acquisition and evaluation of sample novelty.
- *Compensation* - the adaptation by itself is a mechanism how to compensate for gradual changes in data

Because of the reasons mentioned above, the adaptive novelty detection is a promising field of the machine learning for future research.

1.4 Novelty detection implementation challenges

In general, a novelty detection process can by understood as a type of classification. However, the type of the classification used for the novelty detection varies according to the used approach. Some methods dealt with the novelty detection as with a binary classification - the first class is a normal event and the second class is a novel event. Other methods used multiple classes, where one or more classes represents the novel events. Also it is not uncommon to understand the novelty detection as a classification with unknown class or classes.

The issue of novelty detection as a classification is evident. A sufficient training sample is a must for any successful classification. However novelty detection is a task commonly defined without any information about how the novel events should look like. Basically the novelty detection is search for something that is not known, hard to model and difficult to predict.

However, in specific cases it is possible to at least annotate the novel event retrospectively and thus the novelty detector (classifier) can be supervised, or at least some kind of *reinforcement learning*[3] can be applied.

[3]Reinforcement learning is process where some kind of reward function is used instead of an exact input/output pair as a feedback.

Chapter 2

State of the art

The current state of research in the field of novelty detection is presented in this chapter. The bare minimum on theory, implementation and essential idea behind the main novelty detection concepts are presented in section 2.1. The cross-validation concepts used in this thesis and in general are explained in section 2.2. The main approaches and directions in novelty detection field are introduced one by one in section 2.3.

2.1 Novelty detection concepts introduction

The novelty detection may be supervised and also unsupervised. Supervised means, that we have some information how the novelty in the data should look and thus we can train a model to search for this novel events or objects. The term unsupervised novelty detection means, that we do not know what is the novelty in given data-set and we need some method to identify and describe those not common pieces of data. Note that the supervised novelty detection is not too different from an ordinary two-class classification. The classification approach is suitable for any cases, where it is possible to measure whether the finding of the novel event was correct.

Also for some cases of unsupervised novelty detection, it is possible to use a clustering algorithm as a novelty detector. However, this approach could have a problem with insufficient number of anomalies for pre-training of such a clustering model. For this reason, the novelty detection methods are mostly designed in the way

that they measure the distance between the current state and the normal state. The decision whether the distance is big enough follows. If the distance is big enough, the particular state can be considered as a novel state or event.

Common problems associated with novelty detection are: noisy data features; not enough samples for detection; too many normal (and different) states of the system; and insufficient system identification. The other problem may be a difficulty to find out whether the novelty detection works well or not for the given task. This problem is commonly caused by absence of any suitable benchmark. For a lot of applications, only one way how to evaluate novelty detection method exists - human expert advice.

Because of all those issues related to the novelty detection, several different methods of the novelty detection was developed. According to results of those methods, it is possible to say that the success of a given method strongly relies on given task and conditions. There is no universally best solution for case of novelty detection [8]. This statement can be related to the *no free lunch theorem* [9].

Novelty detection methods are commonly separated into statistical based and neural network based. Also it is common to combine something from both approaches in one method. A statistical approach of novelty detection uses statistical properties of the acquired data to decide, whether the data is novel or not. Statistical novelty detection methods could be divided between two groups - parametric approaches and non-parametric approaches. Parametric approaches expect, that distribution of evaluated data is Gaussian in nature. It means, that the data distribution can be modelled just with the data mean and covariance. Non-parametric approaches are more flexible, because they do not have assumptions about the data distribution form. This cause that they are also more computationally expensive [8] in general. However, it is important to note that literature on this topic is not completely united in opinion what methods are parametric and what methods are non-parametric [10].

The example of the most simple statistical approach of novelty detection is box-

plot[1]. Another simple example is histogram[2]. More about usability of this methods for novelty detection is in subsection 2.3.1.

The second category of novelty detection methods is based on artificial neural networks (ANN) [12]. Such methods are heavily used for novelty detection tasks today, because of the recent general popularity of ANN. The ANN based methods have advantages and also disadvantages in comparison to the statistical approaches. Probably the main advantage of the ANN based methods is possibility of online retraining. Commonly discussed disadvantage of the ANN is the huge dependence on the chosen ANN architecture and the complexity of its optimization [13]. If the chosen ANN architecture is too simple, it may have difficulties to learn the system properly. On the other hand, if the architecture is too complicated, it may lose the ability of the generalization that leads to bad performance. For selection of a correct ANN architecture a few approaches exist. The most common and intuitive ones are performance testing while pruning - decreasing complexity, and performance testing while increasing the architecture complexity (also known as constructive algorithms). In general the most common ANN architecture also for novelty detection is multi-layer perceptron (MLP) [14]. The confidence measure of a MLP input patterns is popular novelty indicator. The simplest method how to achieve that is to put a threshold on the ANN output. In other words, the MLP recognize whether the new pattern is know or unknown.

2.2 Cross-validation of novelty detection methods

In matter of cross-validation, a novelty detection can be understand as a binary classification for the purpose of cross-validation. That allows us for using conventional tools, tests and concepts to evaluate the outcome of a novelty detection method. The common way how to systematically describe the process of cross-validation is the construction of the confusion matrix (Table 2.1). The confusion matrix is based

[1]Boxplot [11] (or box-and-whisker plot) is popular plot that uses box (and sometimes whiskers) to display quartiles, median and extreme values of a data sample, invented by J. Tukey.

[2]A histogram is an accurate representation of the distribution of numerical data. It is an estimate of the probability distribution of a continuous variable.

on estimation of four variables:

- *True positive* (TP) - number of successful hits
- *True negative* (TN) - number of correct rejections
- *False positive* (FP) - number of type I error[3]
- *False negative* (FN) - number of type II error[4]

The other metrics that are used for cross-validation are obtained from the derivations of the variables above (TP, TN, FP, FN). The most common derivations are

- *Specificity* (SPE), also known as *true negative rate*

$$\mathrm{SPE} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \tag{2.1}$$

- *Sensitivity* (SEN), also known as *recall*, *hit rate*, or *true positive rate*

$$\mathrm{SEN} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{2.2}$$

- *Precision* (PPV), also known as *positive predictive value*

$$\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{2.3}$$

- *Accuracy* (ACC)

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{2.4}$$

Other tool used for cross-validation of binary classifiers is the receiver operating characteristic (ROC) [15]. The graphical plot of the ROC is called ROC curve and it is probably the most fundamental plot used to illustrate the discrimination threshold of a binary classifier. The ROC curve is obtained by plotting the *true positive rate* (TPR) against the *false positive rate* (FPR). Note that the following relations hold: $\mathrm{TPR} = \mathrm{SEN}, \mathrm{FPR} = 1 - \mathrm{SPE}$. The ROC curve with examples of the optimal, the absolutely random, a good and a bad prediction performance is shown

	Predicted False	**Predicted True**
False Condition	True Negatives (TN)	False Positives (FP)
True Condition	False Negatives (FN)	True Positives (TP)

Table 2.1: Confusion matrix

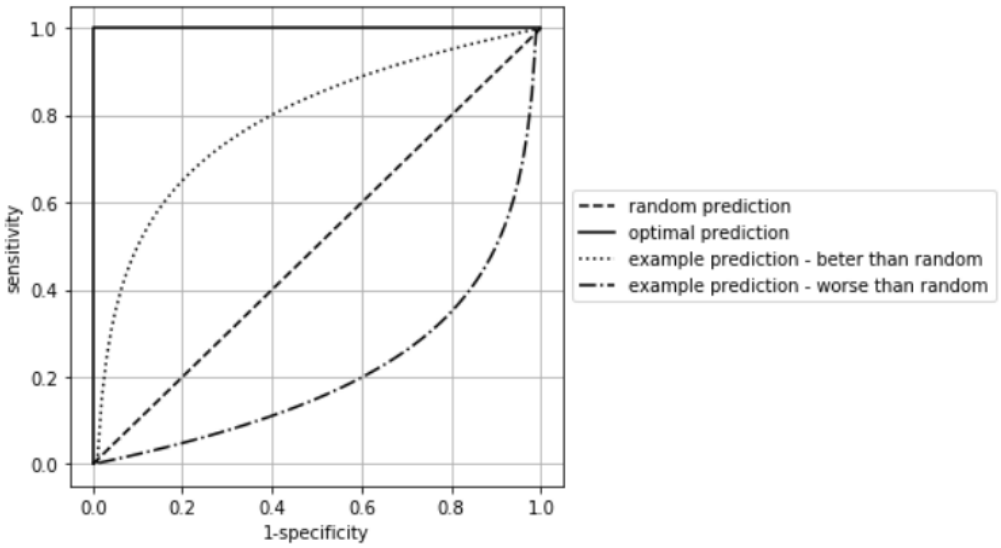

Figure 2.1: The ROC curve with examples of the optimal, the absolutely random one, a good and a bad one prediction performance.

in Figure 2.1. In general, the ROC is obtained by calculating TPR and FPR for changing classification threshold.

Another concept related to the ROC curve is the *area under the roc* (AUROC, often also referred only as AUC). The AUROC is an indicator how well the classifier performs independently on the selected threshold. In other words, the AUROC can be understood as overall performance for all possible threshold settings. The AUROC is commonly used together with maximal accuracy to compare the performance of multiple classifiers. The maximal accuracy (maximal ACC) is accuracy obtained with the optimal threshold. These two metrics (maximal ACC and AUROC) are

[3]A type I error is the incorrect rejection of a true null hypothesis

[4]A type II error is the failure to reject the false null hypothesis

used together because they are not necessary correlated and they describe different aspects of performance. While the AUROC describes the performance of the classifiers independently on a threshold, the maximal ACC highlight the best possible accuracy that can be achieved if the threshold is set correctly. So it is possible that some classifiers score with a maximal ACC, although they have low AUROC and vice versa. However, it is important the keep in mind that the ACC is highly affected by class imbalance. An example is a classifier that predicts false all the time. This classifier can still get high accuracy if a dataset contains many more negative samples than positive ones.

2.3 Main approaches to novelty detection

2.3.1 Hypothesis testing

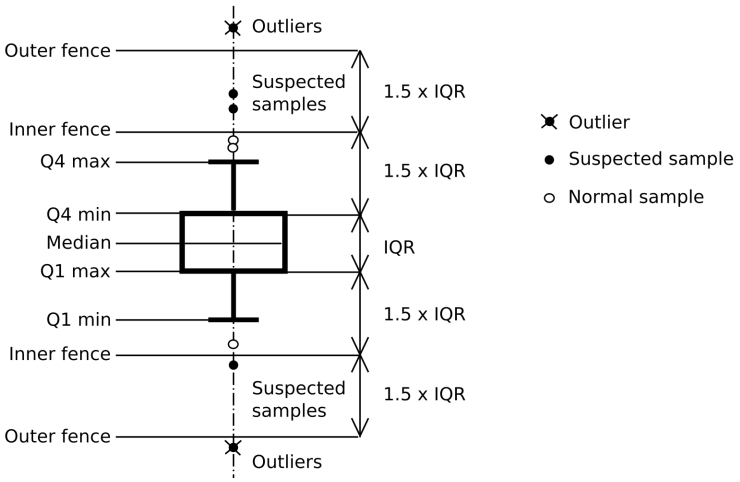

Figure 2.2: The box plot rule for novelty detection (Q = quartile, IQR = interquartile range). In this case the IQR is the span occupied by the second and the third quartile.

Hypothesis testing belongs into the group of parametric approaches. It is simple statistical method commonly used for testing whether the tested data or sample belongs to the same distribution as training data or not. The test popular for this topic is the *Grubbs' test* [16]. The *Grubbs' test* is based on comparison of distance from the test data points and the sample mean. Any data point with this distance

higher than a certain threshold is considered as an outlier. The popular value for the threshold is commonly the value of three standard deviations from the mean value. This test assumes that the training data posses Gaussian distribution and it works only with univariate continuous data. Many variants of this test was proposed later to deal with multivariate data sets, for example [17]. The hypothesis testing approach has been used for detection of damaged beams with *t-test* in study [18]. This method uses subsequent measurements and compare them against the previous values. The results of the study was promising, however the method was tested just on simulated data representing this single problem. In other study [19], the boxplot rule was used to visually localize the outliers in data. The boxplot rule is a commonly used method for outlier detection in unstructured datasets. The inner fence and outer fence are defined in the boxplot and anything out of the fences is considered as an outlier. The position of inner and outer fences is estimated according to the interquartile range (IQR). The boxplot rule usage is displayed in the figure 2.2. More detailed information on the topic of statistical tests for the novelty detection can be found in [20].

2.3.2 Gaussian mixture model

Gaussian mixture models (GMM) [20] is a parametric probabilistic approach. This approach is based on the idea that data are generated from a weighted mixture of Gaussian distributions. Thus it is considered as a parametric approach. Although similar approach - general mixture model - can be based on various different distributions (the gamma distribution, the Poisson distribution, the Student's t distribution), the Gaussian distribution is popular of its convenient analytical properties. The GMM models are used as an estimator of the probability density of the normal data points. The parameters of such a model are generally estimated by a maximum likelihood method or by the Bayesian methods. The novel data points are identified via threshold.

However, in practice the GMM approach (or similar) has a problem with the dimensionality of data. With the high dimensionality, this method needs a very large

number of samples to train a model [8]. Another problem is the correct selection of suitable threshold. Study [21] proposes the GMM for modelling of a text; it uses GMM with Latent semantic analysis representation for novelty detection. This method works with very high dimensional lists of terms and the reported performance is comparable with other state-of-the-art methods. In study [22], the GMM based novelty detection was used for identification of masses in mammograms.

2.3.3 Hidden Markov models

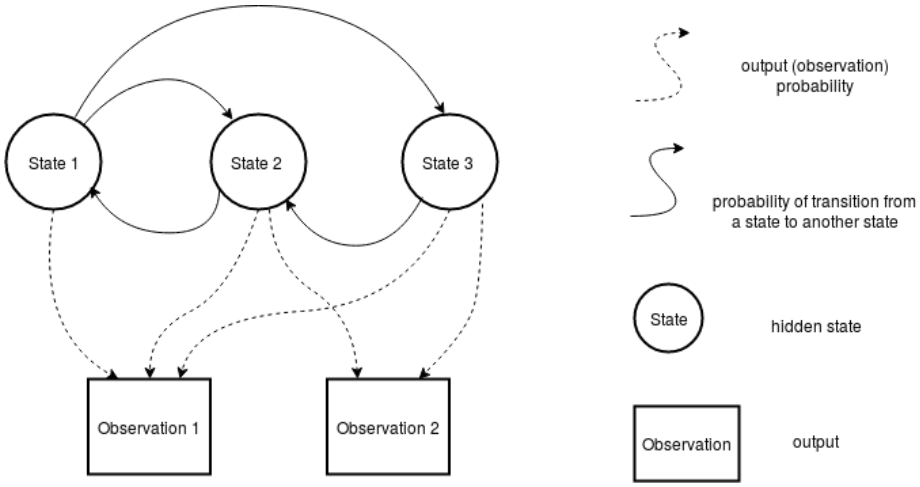

Figure 2.3: An example of Hidden Markov model with three hidden states and two observable outputs.

Hidden Markov models (HMM) [23] are stochastic models for sequential data, and belong into group of parametric approaches. The HMM is build on assumption, that modeled system is process with unobserved hidden states - Markov process. The transition from the hidden states to observable states is done via stochastic process. Every observable state is associated with a set of probability distributions. The change in probabilities of any observable event is compared to a threshold to test for novelty. The hidden Markov models are generally popular for pattern recognition, for example: temporal pattern recognition in bioninformatics, speech, gesture recognition, handwriting recognition and similar tasks. An example of HMM model is shown in Figure 2.3.

In study [24], the HMM based approach was used for intrusion detection in com-

puter security and compared with instance-based learning novelty detection method. The reported accuracy of both methods was similar, but the HMM approach has much lower computational and storage requirements. The HMM based approach for intrusion detection was used also in study [25]. The reported accuracy was also comparable with other methods, while computational cost seems to be a bit lower. Another applications of the HMM based approach for anomaly detection in field of internet security is in [26], [27], [28]. In study [29] was proposed to use the HMM for abnormality in the duration of human daily living activities also with a promising results. The study [30] presents a HMM based framework for anomaly detection in crowd behavior from a video records. The issue of this method is that modeled distribution must be Gaussian in nature.

2.3.4 Support vector machines based approach

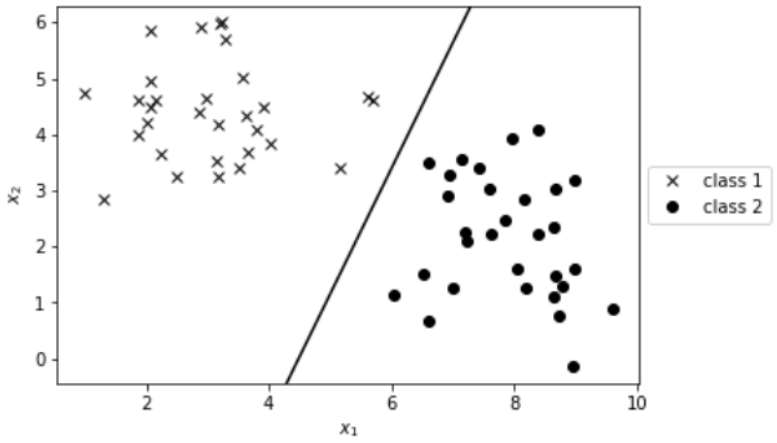

Figure 2.4: An example of the linear SVM binary classification for a data with two features (x_1, x_2). The line represents the border between the classes.

Support Vector Machines (SVM) [31] is a not-probabilistic method [32]. It is designed to work as a binary classifier. In other words the SVM algorithm search for best hyperplane to separate the known classes. An example is in the Figure 2.4. Although the algorithm was originally created as linear classifier, it was later extended with kernel transformation to nonlinear classifier. The kernel transformation can change linearly inseparable task into the task that is linearly separable. As a

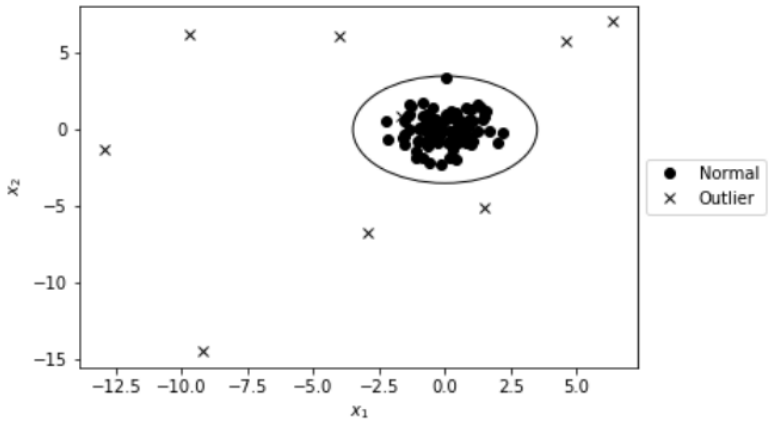

Figure 2.5: An example of the SVM novelty detection. Every data-point laying out of the hypersphere is an outlier.

probabilistic novelty detector the SVM can be used as a classifier for the input vector distribution. However, commonly it is used only as one class classifier - where the one class is the normal class with tight boundary. Anything behind the boundary is classified as novel [33]. An example of such a classification is in Figure 2.5. The major advantages of SVM algorithm implementation are: it works really well with clear margin of separation, it is effective in high dimensional spaces, and it is effective in cases where number of dimensions is greater than the number of samples. On the other hand, SVM are not really good for data with overlapping distributions (small margin between classes).

The one-class SVM approach was studied on artificial and real data with promising results in [33]. In study [34], a method based on SVM for novelty detection in the Electroencephalography (EEG) signal is presented. The targeted novelty in this study is an epileptic seizure. The reported results are not better then other state-of-the-art methods, however the usage of novelty detection as an EEG seizure detector is beneficial, because it does not need supervised pre-learning. In study [35], an algorithm for online novelty detection based on SVM was proposed. This algorithm is processing the data sequentially during training and uses different update equation that has much lower time complexity. Other real-time novelty detection method based on generalized SVM was proposed in [36] for the novelty detection in video surveillance.

Furthermore, support vector data description (SVDD) [37] based approaches are similar to the SVM approaches. The SVDD is a method how to describe a dataset. This description can be used to determine if a new sample is normal or novel. This approach is based on SVM. The data description is spherically shaped boundary around the normal data set. Multiple extensions of SVDD were proposed. Some extensions are based on modifying the margin and/or size of the hypersphere of the data descriptor [38, 39]. Other extension is based on usage of multi-hypersphere data description [40]. This approach was even more stretched with multi-hyper-spheres with different centres and radii in [41]. The experimental results of this extension claim to outperform the original SVDD in all 28 tested datasets. A different extension of the original SVDD focuses on speed of the algorithm by proposing an effecient SVDD [42]. This approach seems to outperform the both one-class SVM and SVDD in speed.

2.3.5 K-nearest neighbour algorithm

K-nearest neighbour algorithm (KNN) [43] is non-parametric statistical method used for classification and regression. The KNN based methods are among the most popular methods for novelty detection nowadays [10]. The KNN algorithm assigns the class for every new sample according to the class of k nearest neighbours. An example is shown in Figure 2.6. The KNN novelty detection approach is based on the assumption that normal data-points have close neighbours, while novel points are located far from those points [44]. The distance of the points is commonly measured with Euclidean or Mahalanobis distance[5]. Other well documented measures that are suitable for KNN algorithm can be found in [23].

The main potential issue of the KNN is a classification output dependency on parameter k (the number of the nearest neighbours). This problem is a case of sensitivity to the local structure of data. This issue can be addressed with weighted sum of the distances from the new data-point to the nearest neighbours. This approach

[5]The Mahalanobis distance [45] is a multi-dimensional generalization of the idea of measuring how many standard deviations away the point is from the mean of the distribution. This concept was introduced by P.C. Mahalanobis in 1936.

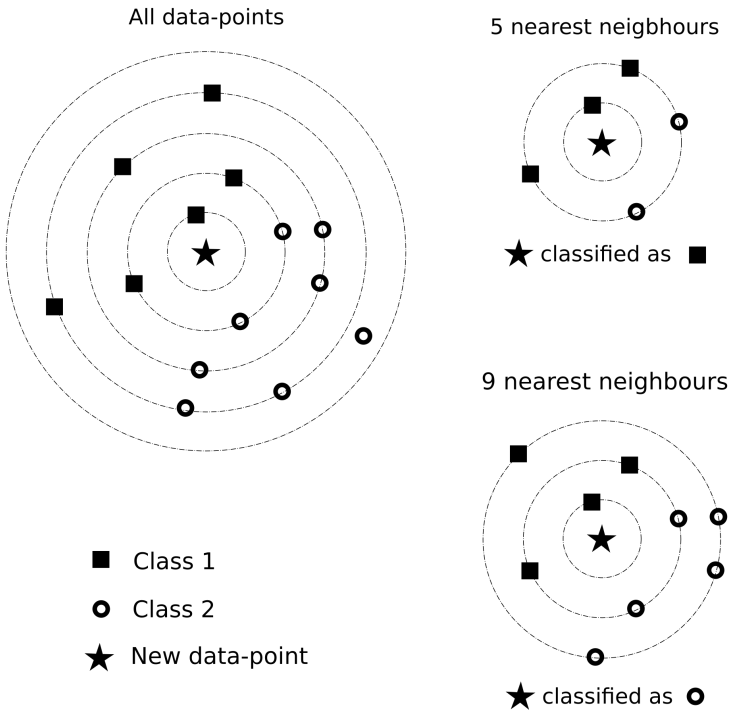

Figure 2.6: An example of the KNN classification. Note that the classification yields a different result for a different number of the nearest neighbours.

was successfully used for outlier detection in study [46]. Another issue of the KNN algorithm is the need to measure the distance from every point to find out the nearest neighbors. However this problem can be reduced with various algorithm extensions.

The KNN based method was used in study [47] for intrusion detection using KNN based classification of short system call sequences. The reported accuracy is comparable with the other methods, but amount of needed computation time seems to be smaller. The study [48] proposes a method for fault detection using the KNN rule for semiconductor manufacturing processes. The obtained results are compared with Principal Component Analysis (PCA) [49] on the same data. Conclusion of the study is that KNN based method works better with given conditions.

However, the KNN technique has problem with huge data-sets, because its evaluation demands much bigger number of computational operations [8].

2.3.6 Neural networks clustering based methods

Clustering is operation of splitting data into groups. For novelty detection it means classification to normal data and novel data. It could be either supervised or unsupervised.

Probably the most common ANN for clustering is self-organizing map (SOM) [50]. As the survey [6] suggests, the SOM is also really popular concept for novelty detection. It is non statistical alternative to clustering algorithms. Therefor, the most common task suitable for SOM is classification of an input patterns. According to the SOM classificator characteristics, the most intuitive way how to use SOM for novelty detection is to use it just as a classifier for the detection whether the given state is normal or novel [51]. Another approach for novelty detection with SOM is monitoring of firing units in the map. For the evaluation of SOM units firing could be used Euclidean distance of map units, or directly unit indexes [52]. Study [53] proposed a novelty detection method based on robust rejection filtering mechanism. For clustering, they used analysis of inter versus intra-cluster distances to find out which cluster represents the novel data. In study [54], the SOM is used to estimate the novelty in features obtained as vector of adaptive parameters of higher order neural units (HONU).

Although the clustering approaches have shown some potentionals in mentined studies, they struggle with the issues of correct segments selection and feature extraction [53] in general.

2.3.7 Reconstruction based approaches

Reconstruction based approaches are methods based on data reconstruction (current or historical sample estimation from reduced features) or prediction. Various ANN architectures or similar learning models could be used for this purposes. Basic idea is that the adaptive model is trained for reconstruction (prediction) of input data. When input data vary from training data, reconstruction error rises up. This method could be used online with forward reconstruction. Big advantage of this approach is simple retraining (online adaptation). During the retraining the increment of adap-

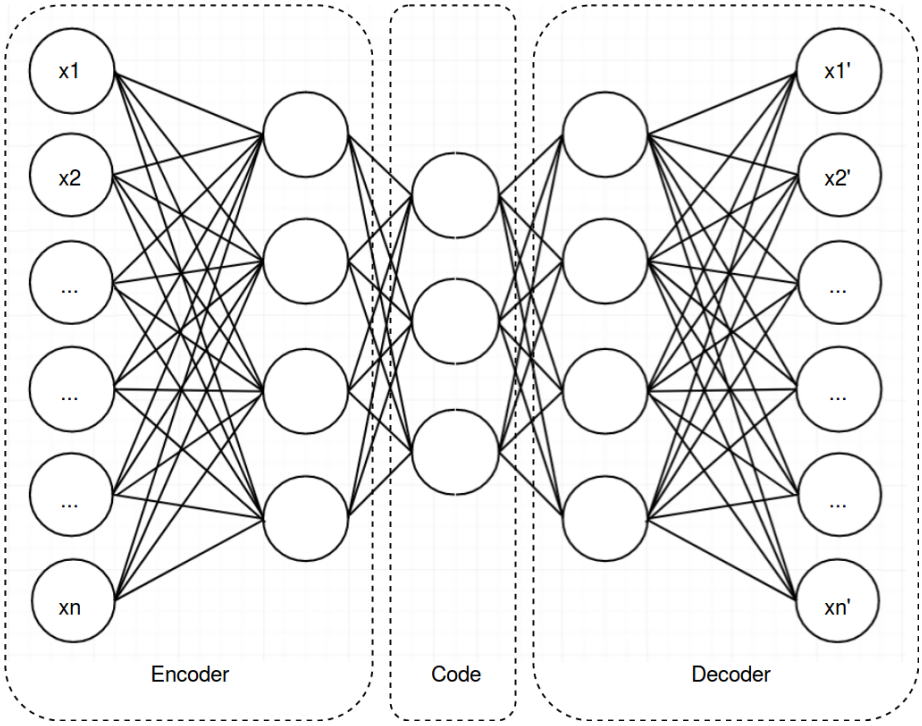

Figure 2.7: The general schema of auto-encoder design. The output (on the right) is reconstructed from the code (in the middle) in order to follow the input (in the left).

tive parameters of the auto-associator could be used as a novelty indicator. These features make the reconstruction based approaches excellent for online processing of data streams.

The adaptive model used for this approach could be as simple as an adaptive filter used in predictive settings. However, the most common data reconstructors are auto-encoders [55] and replicator neural networks (RNN) [56]. Such a network architecture is a model that uses dimensional bottleneck between input and output to filter redundant and incorrect information in training data-set. Particularly the auto-encoders squeeze the input through a hidden layer that has fewer neurons than the input/output layers. This is the way how the network is forced to learn a compressed representation of the data. The architecture of typical auto-encoder is displayed in Figure 2.7. The original idea behind the auto-encoders is based on Elman network [57]. The RNN also squeeze the data through a hidden layer; however, that layer uses a staircase-like activation function. The staircase-like activation function makes the

network compress the data by assigning it to a certain number of clusters (depending on the number of neurons and number of steps). This approach yields somehow different results than plain auto-encoders.

The reconstruction approaches are applicable on various learning algorithms. Applicability of MLP was tested for this purposes in study [58]. This study concludes, that the probabilistic ANN [59] works superior when compared with back-propagation ANN. A recently developed method based on evaluation of data prediction process is Learning Entropy (LE) [60]. Other application of MLP was reported by study [61]. In this study, the distribution of data identification error was evaluated and considered as a novel, if an error was unexpectedly higher than the error from training data. Study [62] tests auto-associators to detect faults according to model residuals. The autoassociative mappings using the kernel based approach [63] and the least squares approach [64] were used in the tested methods. According to the reported results, the kernel based approach seems to work better. Study [65] and study [66] proposes to use RNN [67]. They tested RNN approach on multiple data-sets and report promising results.

Methods based on data reconstruction seems to work well. But they also have the same issues like other ANN methods - it is hard to make a mathematical evidence why it works [61]. Also the computational demands of such algorithms could be overwhelming for some applications.

2.4 Open problems

Major novelty detection methods were presented in this chapter. Most of these methods feature at least one of the following issues:

- A method needs an a-priori information about the novelty and/or healthy data. In other words, the method works as a classifier for known classes.
- A method needs a heavy pre-training and/or has a great time complexity. Thus the method is suitable only for offline use.
- A method heavily relies on statistical attributes of the data. Therefor the

method has a hard time to deal with non-stationary process - data with any kind of concept drift.

Chapter 3

Thesis objectives

As it was introduced in previous section, the machine learning field has currently huge demand for algorithms that can work for data streams produced in real time. According to this demand, the first objective of the thesis is set to:

1. objective - *Development of an adaptive novelty detection method suitable for online data streams processing.* Such a method should be able to re-adapt to new data on the fly without the need for any time expensive re-learing.

The optional but often required feature of novelty detection methods for data streams is computation speed. Not every real-time process use high sampling rate, however a low lag novelty detection is generally beneficial. Because of that reason, the second objective of this study is

2. objective - *Development of a fast adaptive novelty detection method applicable with fast adaptive algorithms.* Such a method should have low time complexity. Thus, it should be suitable for machines with low computational performance.

The fast adaptive algorithms can be for example adaptive filters. The usage of adaptive filters for novelty detection is in principle similar to the usage of neural network auto-associators. The difference is that adaptive filters are simpler, thus

they can run faster and they are easier to implement. That is beneficial because of lower time complexity, but the drawback is the lower abstraction ability of adaptive filters algorithms in comparison to neural networks. However, in some case the lower level of abstraction ability might by desirable. This is because the lower level of abstraction also means lower complexity and thus higher possibility to explain the learning algorithm behavior.

The online data streams processing posses the issue of concept drift and other significant data imbalances that cannot be removed in real-time. Hence the third objective of this thesis is set to:

3. objective - *Development of an adaptive novelty detection method robust against concept drift and non-stationary data.* Data trends and drifts are common and it is hard to deal with them in a real-time processing. Therefore the proposed algorithm should be able to compensate for data drifts on the fly.

In the next chapters, derivation, implementation and experimental analysis of the adaptive novelty detection method that should accomplish all objectives of this thesis are presented.

Chapter 4

Developed method

In this chapter, the developed method called Error and Learning Based Novelty Detection (ELBND) is introduced and explained. The general idea and derivation of the main detection rule is presented in section 4.1. The particular implementation of the ELBND for specific adaptive filters is demonstrated in section 4.2. The final notes on derivation and implementation of this method are summarized in section 4.3.

4.1 Method description

The proposed method of novelty detection utilizes the adaptive parameters of a learning model and its error. This method could be implemented on various supervised adaptive models (tracking adaptive algorithms). The idea behind this method is based on assumption, that the model error and the adaptive parameters of the model carry a different information about novelty of data, although both features are correlated. On this account, the proposed method is called Error and Learning Based Novelty Detection (ELBND).

In this work, various types of adaptive filters are used as the base for adaptive models. An adaptive filter is a system with a linear filter that has a transfer function controlled by variable parameters to adjust those parameters according to an optimization algorithm. The scheme of an adaptive filter function is displayed in figure 4.1

The output of adaptive filter or of any similar adaptive model could be described

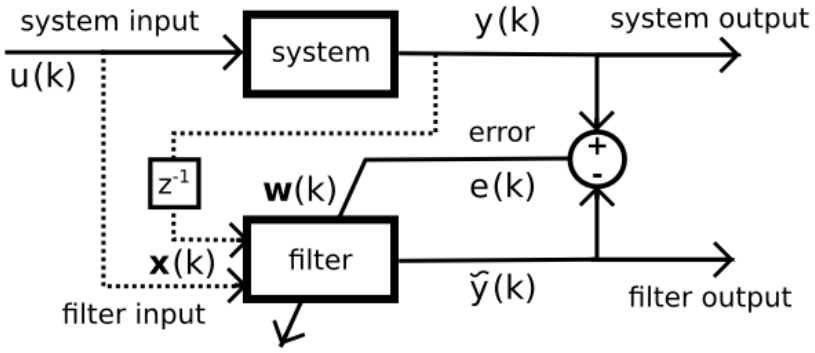

Figure 4.1: The schema of an adaptive filter function.

with the following equation

$$\tilde{y}(k) = w_1(k) \cdot x_1(k) + ... + w_n(k) \cdot x_n(k) = \sum_{i=1}^{n} w_i x_i = \mathbf{w}^T(k) \cdot \mathbf{x}(k), \tag{4.1}$$

where k is discrete time index, $\tilde{y}(k)$ is output (filtered) signal, $\mathbf{w}$ is vector of adaptive weights, $\mathbf{x}$ is input vector and $(.)^T$ denotes the transposition. The initial values of adaptive weights (adaptive parameters) $\mathbf{w}$ are usually set to all zeros, or alternatively to random numbers (normal distribution, zero mean value). The $\mathbf{x}(k)$ is input vector made from input data

$$\mathbf{x}(k) = [x_1, ..., x_n], \tag{4.2}$$

where n is the size of input vector. The input vector can be augmented with bias $(= 1)$ as follows

$$\mathbf{x}(k) = [1, x_1, ..., x_n]. \tag{4.3}$$

The bias should mimic the bias in neural units. In practice, this bias can compensate offsets and similar data imbalances. In case where only input data is the history of the target signal, the input vector can be formed as follows

$$\mathbf{x}(k) = [1, y(k-n-1), ..., y(k-1)], \tag{4.4}$$

where y is the measured signal. The mentioned error e of the adaptive filter is calculated as

$$e(k) = y(k) - \tilde{y}(k). \tag{4.5}$$

The second input of the method is the increment of adaptive weights defined as

$$\Delta\mathbf{w}(k) = \mathbf{w}(k+1) - \mathbf{w}(k). \tag{4.6}$$

The method of the increment $\Delta\mathbf{w}(k)$ estimation depends on chosen learning algorithm. The proposed way how to combine the parameters $\Delta\mathbf{w}(k)$ and error $e(k)$ to obtain the descriptor of novelty in given sample can be described as follows

$$\mathbf{nd}(k) = \Big|e(k) \cdot \Delta\mathbf{w}(k)\Big|. \tag{4.7}$$

The novelty descriptor $\mathbf{nd}(k)$ is vector of coefficients describing how much novelty is encounter with individual weights $\Delta\mathbf{w}(k)$.

For some applications it could be desirable to describe novelty in data just with single value for every sample. As a good practice how to achieve that, it is reduction of this vector $\mathbf{nd}(k)$ to scalar as follows

$$nd(k) = \max(\mathbf{nd}(k)). \tag{4.8}$$

However, other function than max might be used.

4.2 Methods of implementation

4.2.1 LMS adaptive filter

The classical least means squares algorithm (LMS) [68] is stochastic gradient descent method. It is probably the most common algorithm for adaptive filters. The LMS weights adaptation could be described as follows

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \Delta\mathbf{w}(k), \tag{4.9}$$

where $\Delta\mathbf{w}(k)$ is

$$\Delta\mathbf{w}(k) = \mu \cdot e(k) \cdot \frac{\partial y(k)}{\partial \mathbf{w}(k)} = \mu \cdot e(k) \cdot \mathbf{x}(k), \tag{4.10}$$

where μ is the learning rate (step size) and The general stability criteria of LMS [68] stands as follows

$$|1 - \mu \cdot \mathbf{x}(k)^T \cdot \mathbf{x}(k)| \leq 1. \tag{4.11}$$

The novelty detection could be done as follows

$$\mathbf{nd}(k) = \Big|e(k) \cdot \Delta\mathbf{w}(k)\Big| = \Big|e(k)^2 \cdot \mathbf{x}(k) \cdot \mu\Big|. \tag{4.12}$$

4.2.2 NLMS adaptive filter

The normalized least mean squares (NLMS) [68] adaptive filter is extension of LMS adaptive filter. The NLMS adaptation rule could be described as follows

$$\Delta\mathbf{w}(k+1) = \frac{\mu}{\epsilon + \mathbf{x}(k)^T \cdot \mathbf{x}(k)} \cdot \mathbf{x}(k) \cdot \mathbf{w}(k) = \eta \cdot \mathbf{x}(k) \cdot \mathbf{w}(k), \tag{4.13}$$

where ϵ is a constant (regularization term) introduced to preserve stability for inputs close to zero [69]. The model is stable if

$$0 \leq \mu \leq 2 + \frac{2\epsilon}{\mathbf{x}(k)^T \cdot \mathbf{x}(k)}, \tag{4.14}$$

or in case without regularization term ϵ

$$\mu \in < 0, 2 > . \tag{4.15}$$

With the NLMS adaptive filter the novelty in data could be calculated as follows

$$\mathbf{nd}(k) = \Big| e(k) \cdot \Delta\mathbf{w}(k) \Big| = \Big| e(k)^2 \cdot \mathbf{x}(k) \cdot \eta \Big| = \left| \frac{e(k)^2 \cdot \mathbf{x}(k) \cdot \mu}{\epsilon + \mathbf{x}(k)^T \cdot \mathbf{x}(k)} \right|. \tag{4.16}$$

4.2.3 LMF adaptive filter

The least mean fourth algorithm (LMF) [68] is slight modification of the LMS algorithm. The LMF weights adaptation $\Delta\mathbf{w}(k)$ is calculated as follows

$$\Delta\mathbf{w}(k) = \mu \cdot e(k)^3 \cdot \frac{\partial y(k)}{\partial \mathbf{w}(k)} \ = \mu \cdot e(k)^3 \cdot \mathbf{x}(k), \tag{4.17}$$

The ELBND is then calculated as follows

$$\mathbf{nd}(k) = \Big| e(k) \cdot \Delta\mathbf{w}(k) \Big| = \Big| e(k)^4 \cdot \mathbf{x}(k) \cdot \mu \Big|. \tag{4.18}$$

According to the (4.18) the ELBND emphasize the error of adaptive filter more with the LMF than the plain LMS.

4.2.4 NLMF adaptive filter

The normalized least mean fourth (NLMF) is often used because it has greater ability to suppress noise than NLMS adaptive filter according to study [70]. On the other

hand, it is much harder to enforce stability of the NLMS filter than NLMS filter [71, 72]. The NLMF adaptation [68] is similar to NLMS adaptation. The vector of adaptive weights of a NLMF filter $\mathbf{w}$ is done according to the rule

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \Delta\mathbf{w}(k) = \mathbf{w}(k) + \eta(k)\mathbf{w}(k)e(k)^3, \tag{4.19}$$

where $\eta(k)$ has the same meaning like in (4.13). With the NLMF adaptive filter the novelty in data could be calculated as follows

$$\mathbf{nd}(k) = \left|e(k) \cdot \Delta\mathbf{w}(k)\right| = \left|e(k)^3 \cdot \mathbf{x}(k) \cdot \eta\right| = \left|\frac{e(k)^3 \cdot \mathbf{x}(k) \cdot \mu}{\epsilon + \mathbf{x}(k)^T \cdot \mathbf{x}(k)}\right|. \tag{4.20}$$

4.2.5 GNGD adaptive filter

The generalized normalized gradiend descend (GNGD) adaptive filter [69] is an extension of the NLMS adaptive filter. The adaptive weights of a GNGD filter $\mathbf{w}$ are adapted according to the same rule as NLMS. The difference is in parameter $\eta(k)$. The adaptive learning rate (step size) $\eta(k)$ is estimated in similar way like for NLMS or NLMF, however the regularization term ϵ is obtained in the way that follows

$$\epsilon(k) = \epsilon(k-1) - \rho\mu\frac{e(k) - e(k-1)\mathbf{x}^T(k)\mathbf{x}(k-1)}{(||\mathbf{x}(k-1)||^2 + \epsilon(k-1))^2}, \tag{4.21}$$

where the ρ is a custom parameter. As proposed in [69] the GNGD the method should be robust if the parameter ρ is set to small (< 1) positive number. The resulting GNGD formula for novelty detection can be combined from (4.21) and (4.16).

4.2.6 RLS adaptive filter

Other algorithm what could be used for the novelty detection is Recursive least squares (RLS) [68]. For this method the adaptive weights are calculated as follows

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{R}^{-1}(k)\mathbf{x}(k)e(k) \tag{4.22}$$

where the matrix $\mathbf{R}^{-1}$(k) is inverse of the auto-correlation matrix with size $n \times n$, where n is number of adaptive weights $\mathbf{w}$(k). The $\mathbf{R}^{-1}$(k) matrix is obtained as

follows

$$\mathbf{R}^{-1}(k) = \frac{1}{\gamma}\left(\mathbf{R}^{-1}(k-1) - \frac{\mathbf{R}^{-1}(k-1)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{R}^{-1}(k-1)}{\gamma + \mathbf{x}^T(k)\mathbf{R}^{-1}(k-1)\mathbf{x}(k)}\right). \tag{4.23}$$

The initial value of matrix $\mathbf{R}^{-1}$ is set as follows

$$\mathbf{R}^{-1}(0) = \frac{1}{\delta}\mathbf{I} = \begin{bmatrix} \frac{1}{\delta} & & \\ & \ddots & \\ & & \frac{1}{\delta} \end{bmatrix}, \tag{4.24}$$

where initialization parameter δ stands for small positive constant. According to the adaptation rule (4.22), we can describe proposed novelty detection method with RLS algorithm as follows

$$\mathbf{nd}(k) = \left|e(k) \cdot \Delta\mathbf{w}(k)\right| = \left|e^2(k)\mathbf{R}^{-1}(k)\mathbf{x}(k)\right|. \tag{4.25}$$

4.2.7 Individual learning rate LMS/NLMS adaptive filter

This filter [mc6] is extension of LMS or NLMS adaptive filter. The extension is a replacement of the scalar learning rate μ with vector of learning rates $\boldsymbol{\mu}$. With this modification the increment of adaptive weights $\Delta\mathbf{w}(k)$ is

$$\Delta\mathbf{w}(k) = \boldsymbol{\mu} \cdot e(k) \cdot \mathbf{x}(k) = [\mu_1 e(k)x(k)_1, ..., \mu_n e(k)x(k)_n]^T, \tag{4.26}$$

In this case the general stability criteria stands as follows

$$|1 - \boldsymbol{\mu} \cdot \mathbf{x}(k)^T \cdot \mathbf{x}(k)| \leq 1. \tag{4.27}$$

And the novelty detection could be done as follows

$$\mathbf{nd}(k) = \left|e(k) \cdot \Delta\mathbf{w}(k)\right| = \left|e(k)^2 \cdot \mathbf{x}(k) \cdot \boldsymbol{\mu}\right|. \tag{4.28}$$

The NLMS adaptation rule with individual learning rates could be described as follows

$$\Delta\mathbf{w}(k+1) = \frac{\boldsymbol{\mu}}{\epsilon + \mathbf{x}(k)^T \cdot \mathbf{x}(k)} \cdot \mathbf{x}(k) \cdot \mathbf{w}(k), \tag{4.29}$$

where ϵ is a constant (regularization term) introduced to preserve stability for inputs close to zero [69]. The model is stable if

$$0 \leq \mu \leq 2 + \frac{2\epsilon}{\mathbf{x}(k)^T \cdot \mathbf{x}(k)}, \tag{4.30}$$

or in case without regularization term ϵ

$$\mu \in < 0, 2 > . \tag{4.31}$$

With the NLMS adaptive filter the novelty detection could be done as follows

$$\mathbf{nd}(k) = \left| e(k) \cdot \Delta \mathbf{w}(k) \right| = \left| e(k)^2 \cdot \mathbf{x}(k) \cdot \boldsymbol{\eta} \right| = \left| \frac{e(k)^2 \cdot \mathbf{x}(k) \cdot \boldsymbol{\mu}}{\epsilon + \mathbf{x}(k))^T \cdot \mathbf{x}(k)} \right|. \tag{4.32}$$

4.2.8 Online centered NLMS adaptive filter

The Online Centered NLMS (OCNLMS) adaptive filter is extension of NLMS filter for better convergence with high offset data. This modification was proposed in [mc7]. The main idea behind this modification is data common data transformation for improving the condition number of input data matrix $\mathbf{x}$. This transformation is commonly noted as z-score

$$y_t(k) = \frac{y(k) - \bar{y} \cdot \vec{1}}{\sigma_y}, \tag{4.33}$$

where $\bar{y}$ is mean value of y, σ_y is standard deviation of y and $\vec{1}$ is n sample length vector of all ones. The result of transformed signal filtering $\tilde{y}_t$ could be transformed back as simple as

$$\tilde{y}(k) = (\tilde{y}_t(k) \cdot \sigma_y) + \bar{y} \cdot \vec{1}. \tag{4.34}$$

Filter with normalized data could be defined according to (4.1) as follows

$$\tilde{y}_t(k) = \mathbf{w}_t^T(k) \cdot \mathbf{x}_t(k), \tag{4.35}$$

where $\mathbf{x}_t(k)$ is input vector build from transformed data y_t according to (4.2) and $\mathbf{w}_t(k)$ is set of parameters of adaptive filter for transformed data. From (4.33) and (4.35) is possible to obtain

$$\frac{\tilde{y}(k) - \bar{y}}{\sigma_y \cdot \vec{1}} = \mathbf{w}_t^T(k) \cdot \left(\frac{\mathbf{x}(k) - \bar{y} \cdot \vec{1}}{\sigma_y} \right) \tag{4.36}$$

that could be simplified to

$$\tilde{y}(k) = \mathbf{w}_t^T(k) \cdot (\mathbf{x}(k) - \bar{y} \cdot \vec{1}) + \bar{y}. \tag{4.37}$$

The adaptation rule for such an adaptive filter could be obtained in the same way as filter equation (4.37) from (4.10) and (4.33) as follows

$$\Delta \mathbf{w}_t(k) = \frac{\mu}{\sigma_y^2} \cdot e(k) \cdot (\mathbf{x}(k) - \bar{y} \cdot \vec{1}). \tag{4.38}$$

This is still not beneficial for online filtering, because of the need to know mean value for all the data, what is impossible during real time filtering. Because of that, it is proposed to substitute the $\bar{y}(k)$ with mean value of input vector $\bar{\mathbf{x}}(k)$ and σ_y with σ_x. These parameters of input vector could obtain for every single sample just from vector $\mathbf{x}(k)$. That means that the input vector will be centered

$$\mathbf{x}_c(k) = \mathbf{x}(k) - \bar{\mathbf{x}}(k) \cdot \vec{1}. \tag{4.39}$$

This usability suggestion is based on following assumptions

$$\bar{y} \approx \bar{\mathbf{x}}(k) \wedge \sigma_y \approx \sigma_x. \tag{4.40}$$

Now the equation for online centered adaptation looks like

$$\Delta \mathbf{w}_t(k) = \frac{\mu}{\sigma_y^2} \cdot e(k) \cdot \mathbf{x}_c(k), \tag{4.41}$$

and the filter equation stands as follows

$$\tilde{y}(k) = \mathbf{w}_t^T(k) \cdot \mathbf{x}_c(k) + \bar{\mathbf{x}}(k). \tag{4.42}$$

The general stability criteria can be obtained from (4.41) and (4.39) as

$$|1 - \frac{\mu}{\sigma_y^2} \cdot \mathbf{x}_c(k)^T \cdot \mathbf{x}_c(k)| \leq 1. \tag{4.43}$$

The NLMS algorithm is already using the learning rate normalization (4.13) according to power of input. For that reason there is no need to normalize the learning rate furthermore according to power σ_x. This simplification decreases the error caused by $\sigma_x \neq \sigma_y$. Finally the proposed learning rule could be described as follows

$$\Delta \mathbf{w}(k+1) = \frac{\mu}{\epsilon + \mathbf{x}_c(k)^T \cdot \mathbf{x}_c(k)} \cdot \mathbf{x}_c(k) \cdot \mathbf{w}(k). \tag{4.44}$$

For this filter the novelty in data could be calculated as follows

$$\mathbf{nd}(k) = \left| e(k)^2 \cdot \mathbf{x}_c(k) \cdot \eta \right| = \left| \frac{e(k)^2 \cdot \mathbf{x}_c(k) \cdot \mu}{\epsilon + \mathbf{x}_c(k)^T \cdot \mathbf{x}_c(k)} \right| = \left| \frac{e(k)^2 \cdot (\mathbf{x}(k) - \bar{x}(k)) \cdot \mu}{\epsilon + (\mathbf{x}(k) - \bar{x}(k))^T \cdot (\mathbf{x}(k) - \bar{x}(k))} \right| \tag{4.45}$$

4.3 Method implementation overview

In this subsection, the overview of method implementations for various adaptation rules is presented. Novelty detection rules for all methods described in this chapter are in Table 4.1. The most efficient (the lowest time complexity) is the LMS algorithm. The most complicated one is the RLS algorithm. However it is not possible to conclude that the most sophisticated algorithms yields the best results. For some cases the best results can by achieved with simpler adaptation algorithms due to the multiple reasons.

In general, the proposed adaptive novelty detection method can by implemented for every adaptive model with measurable error, if the used adaptive rule features adaptive parameters. In other words, the ELBND algorithm is not limited to the mentioned algorithms. Some other algorithms that could be used for implementation of the proposed method are: sign-sign least-mean-squares (SSLMS), normalized sign-sign least-mean-squares (NSSLMS) and affine projection (AP).

Filter	$\mathbf{nd}(k) =$
LMS	$\left\|e(k)^2 \cdot \mathbf{x}(k) \cdot \mu\right\|$
LMF	$\left\|e(k)^4 \cdot \mathbf{x}(k) \cdot \mu\right\|$
NLMS	$\left\|\frac{e(k)^2 \cdot \mathbf{x}(k) \cdot \mu}{\epsilon + \mathbf{x}(k))^T \cdot \mathbf{x}(k)}\right\|$
NLMF	$\left\|\frac{e(k)^3 \cdot \mathbf{x}(k) \cdot \mu}{\epsilon + \mathbf{x}(k))^T \cdot \mathbf{x}(k)}\right\|$
RLS	$\left\|e^2(k)\mathbf{R}^{-1}(k)\mathbf{x}(k)\right\|$
iNLMS	$\left\|\frac{e(k)^2 \cdot \mathbf{x}(k) \cdot \boldsymbol{\mu}}{\epsilon + \mathbf{x}(k))^T \cdot \mathbf{x}(k)}\right\|$
ocNLMS	$\left\|\frac{e(k)^2 \cdot (\mathbf{x}(k) - \bar{x}(k)) \cdot \mu}{\epsilon + (\mathbf{x}(k) - \bar{x}(k))^T \cdot (\mathbf{x}(k) - \bar{x}(k))}\right\|$

Table 4.1: Novelty detection rule for different learning algorithms.

Chapter 5

Experimental results

Experimental analysis is the key procedure how to make a discovery or to validate a hypothesis. All published experimental results related to methods introduced in previous chapter are presented in this chapter. This chapter displays the potentials of the proposed novelty detection method.

This chapter is organized as follows: in section 5.1 results related to biomedical data processing are presented, in section 5.2 the experiments featuring concept drift and their results are described, and in the section 5.3 findings from other applications or publications are presented.

5.1 Nonstationary biomedical data

Biomedical sciences are a set of applied sciences derived from natural science dealing with healthcare or public health. The data processing requests by biomedical science researches are mainly related to signals measured on human body, for example: electroencephalography (EEG) [73], magnetoencephalography (MEG) [74] and electrocardiography (ECG) [75]. These signals are naturally complex and non-stationary. However, in cases where only short segments (less than 30 seconds) of reasonably normal signal are used, the signal is considered stationary. Because of the non-stationary and complexity, the biomedical data processing is a challenging task. In this subsection, there are introduced two studies of the proposed novelty detection method use for biomedical data processing - the detection of perturbations

in ECG and the classification of Alzheimer's disease from EEG signal. These two studies should demonstrate the abilities of ELBND to deal with non-stationary and offseted real-time signal.

5.1.1 Perturbation detection in ECG

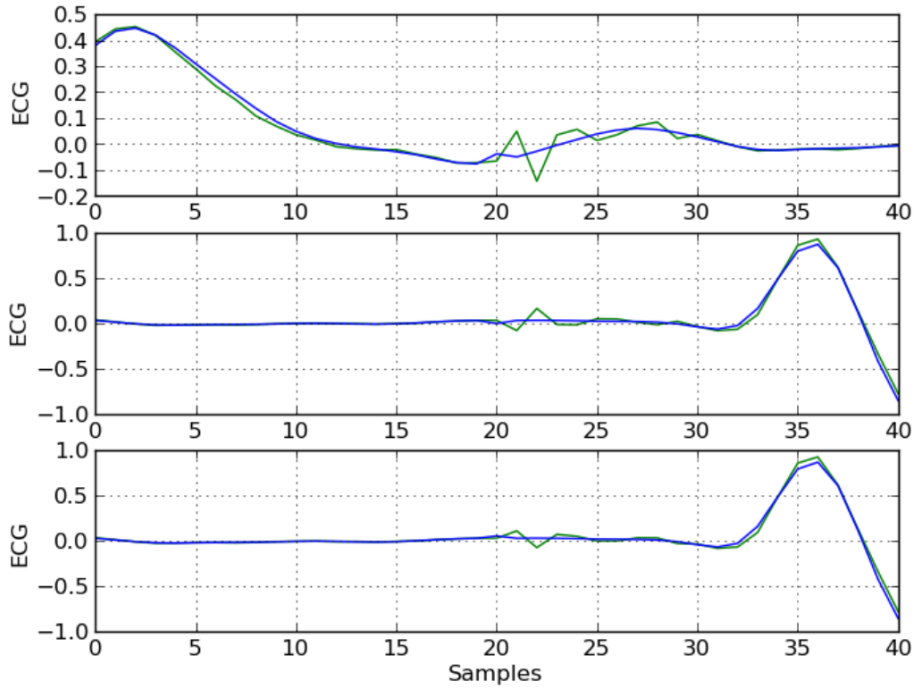

Figure 5.1: Details of prediction in areas of introduced perturbations in artificial ECG without noise - the disturbed line is the prediction. The plot is adopted from study [mc1].

This study was presented in [mc1]. The goal of this study was detection of artificial perturbations in ECG signal. The detection of perturbations can be considered as the value based novelty detection. The ECG measurement is a process of recording the electrical activity of the heart over a period of time using electrodes placed on the skin. A various perturbations can occur in this signal naturally as the ECG artifacts. Such perturbations might decrease the accuracy of further ECG processing or even make it completely impossible. In cases where the perturbations are small, it can be difficult to remove them with conventional methods or by human operator. Because of this reason, the novelty detection can be a simple way how to detect such

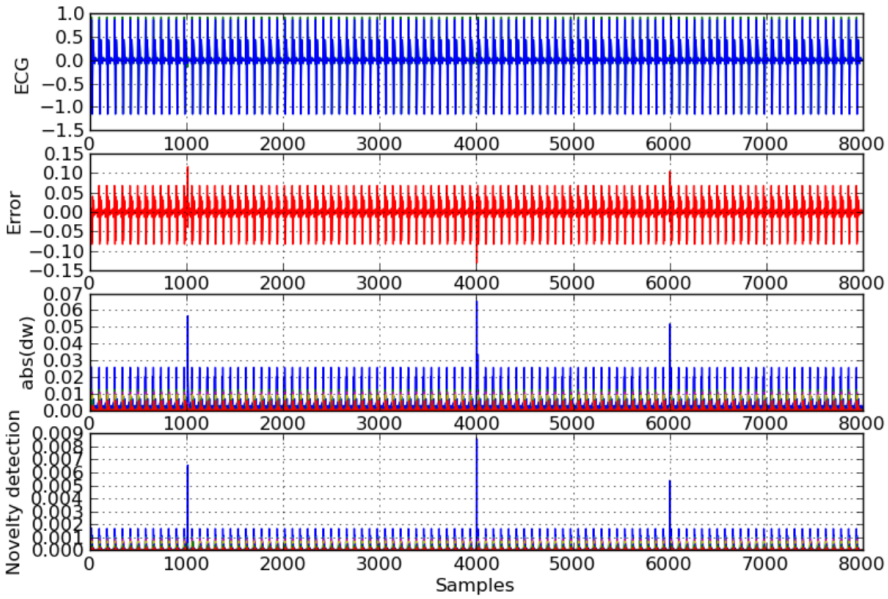

Figure 5.2: Novelty Detection used on artificial ECG signal. The plot is adopted from study [mc1].

artifacts in ECG and report them for further processing.

This study uses artificial and also real ECG signal to demonstrate the idea and to validate its usefulness. The real measured signal (and also the artificial one) has sampling frequency of 256 samples per second (sps). The noise was added to the artificial signal, to highlight the ability of the proposed novelty detection method in detection of unexpected samples within the data. The predictive model features lower prediction accuracy comparative to data without noise. The reason why was this method tested on an artificial signal is to emphasize how well the detection works on perturbed data if the signal does not contain any complicated phenomena. The artificial ECG time series used in this work was created by repeating pattern of a real ECG signal. Thus, this artificial time series is an ideally periodic signal. The used adaptation rule was NLMS applied for linear neural unit (LNU) predictor.

5.1.1.1 Artifical data

The used artificial data has 8000 samples. First 200 samples was used for pre-training. For sufficient training, 100 epochs was enough. In Figure 5.1, there are shown the details of all perturbations included in the artificial ECG signal with-

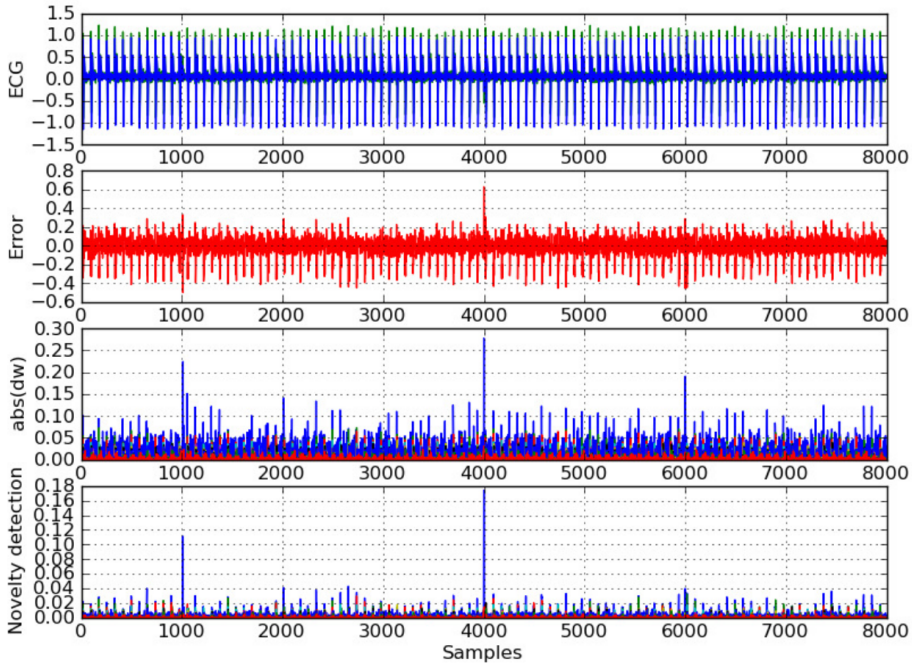

Figure 5.3: Novelty Detection used on artificial ECG signal with noise. The plot is adopted from study [mc1].

out noise. The size of the introduced perturbation was 0.03 mV. As we can see in Figure 5.1, these perturbations are small in comparison with the amplitude of the signal. Looking at the behavior of the used predictive model, it is possible to see how to model re-learn immediately when the prediction error and weight adaptation increases (5.2). The return of the predictive model to previous prediction accuracy takes approximately 20 following samples. In Figure 5.2, it is displayed the prediction error in specific places of a single period. These errors are caused by insufficient prediction ability of the simple, linear predictive model. Furthermore, in Figure 5.2, these errors are not detected as new data by the applied novelty detection method. Figure 5.3, shows the simulation of the artificially created ECG with the addition of noise. Here again, three perturbations were introduced to the data. These perturbations are located on the same positions as the signal without noise. This introduced noise was generated via a generator of pseudo-random numbers, composed as a vector of random numbers in range from 0 to 0.01.

Figure 5.3 displays the artificial signal contaminated with noise. Here in the first graph, it is possible to see the difference between the signals within the region of

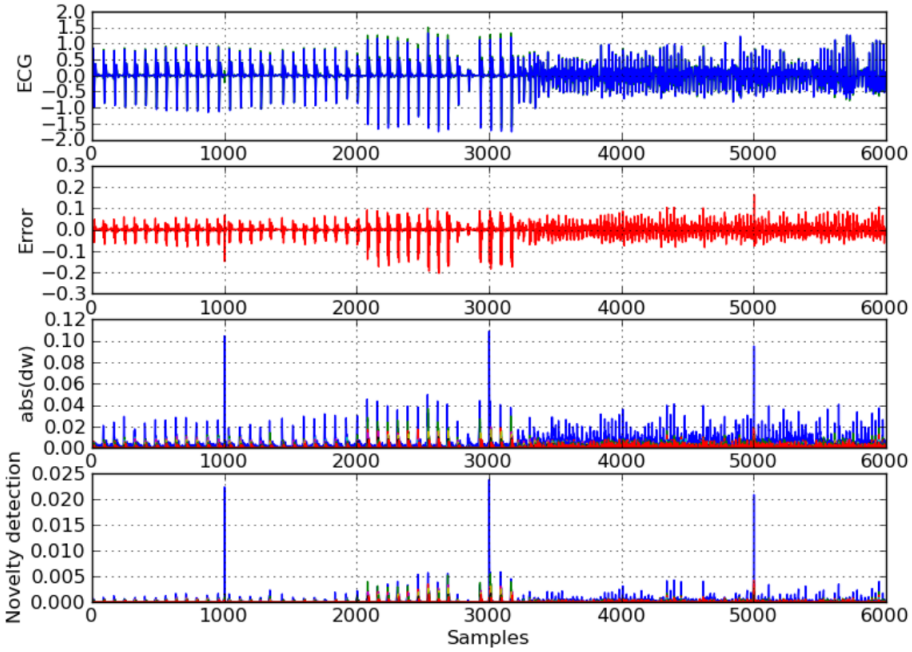

Figure 5.4: Novelty Detection used on real measured ECG signal. The plot is adopted from study [mc1].

peaks of the amplitude. For the used predictive model, it is much more difficult to learn the pattern of the signal with added noise in comparison to that without noise. The errors and absolute values of the weight increments (Figure 5.3) are not entirely dependent on the periodicity of the signal. In the plot of error on the same Figure, it is not possible to see the perturbations clearly, as in the plot of prediction error without noise on the Figure 5.2. In the graph of the absolute error of weight increments we can more evidently see the location of the perturbations. Furthermore, on the graph of novelty detection (Figure 5.3) the perturbation locations are even more evidently seen and, this is because a huge part of the models periodic errors, are filtered. Furthermore, these errors have no impact on the detection of unexpected samples in the data. The adaptive model again immediately reacts to the introduced perturbation and tries to relearn the data signal. The time in which the model needs for regaining normal accuracy after meeting a perturbation is hard to estimate, because the prediction error is highly dependent on the noise level present in the data.

5.1.1.2 Real measured data

The second used signal is real measured ECG signal from the Yoshizawa-Sugita Lab (formerly Yoshizawa-Homma Lab), Tohoku university. The used time series was measured by an internal cardio-defibrillator with frequency of 256Hz. This signal was chosen because it contains spontaneous ventricular tachycardia, which is a rare phenomenon to measure. The used novelty detection method works in both in the healthy section of ECG and also in the arrhythmic section of this signal. The same method and implementation was used for this signal like for the signals used before (the artificial ECG signal). The size of data chosen for learning of the predictive model is 1000 samples. In order to achieve sufficient pre-learning of the used neural unit with such amount of data, less than 500 epochs is enough for achieving optimal results. Usage of more epochs does not improve the accuracy a in significant way. Figure 5.4 shows which part is the healthy ECG signal and which part represents the arrhythmia. In the introduced novelty detection, it is possible to detect the start of the arrhythmia signal, approximately 1000 samples before the arrhythmia is introduced. The shape of the period before the arrhythmia, looks the same but the scale of amplitudes starts changing. In the first graph of Figure 5.4, it is possible to see how the measured signal is practically identical with the predicted signal. The included perturbations are not clearly seen in this graph. However, these perturbations are located in the samples of discrete time 1000, 3000, 5000. On the graph of the prediction error, it is possible to see the perturbations, and even more so in the graph of absolute values of adaptive weights increments. However, looking on the graph of novelty detection, these perturbations are even more evidently pronounced. Moreover, the periodic error are suppressed in region of the healthy ECG and arrhythmia signal. It is important to notice that the suppressing of the periodic error, is not that dominant in the onset of arrhythmia unlike in other parts of the data. The included perturbations are significantly small in size (0.04 – approximately 2% of the amplitude of healthy ECG signal). Furthermore, the adaptive model immediately reacted with re-learning of the ECG data and achieved the previous model accuracy. This re-learning takes about 5-10 next samples.

5.1.1.3 Summary

This experiments demonstrate that the proposed novelty detection method is capable to highlight perturbations in the ECG data, even if the data are contaminated with noise. The simulation was performed on a personal computer and measured speed was higher than what is sampling of ECG signal. In other words, this implementation can be used in real time.

5.1.2 Alzheimer's disease classification

This section is based on study [mc8]. In this case the novelty detection method was used for extraction of features from EEG signal. The features were used for classification of patients. Simple classification method was used to determine how usable the extracted features are for Alzheimer's disease detection. As the data records of EEG obtained from hospital were used. These records are manually selected section with no artifacts. Data selection contains records from 220 anonymous patients. From that selection, 110 patients match the clinical criteria of dementia and the rest are normal. Every patient has 2 to 5 manually selected records with length 90s or less.

The EEG signal history was used for adaptation of LNU predictor. The tested history windows for prediction of one sample were really short (4 samples back and 9 samples back). This history cannot contain complete information about signal dynamics. That cause significant prediction inaccuracy. But such inaccuracy is not an issue for this method. Actually the proposed method works better with less precise simple model, than with more complicated models what we have tried in this case. This fact could be an advantage in case of implementation for real-time usage. Smaller size of input vector means less calculations for one sample prediction. Interesting thing is that the correlation coefficient between real measured EEG signal and prediction output is 0.3856 (4 samples history as input) and even just 0.3098 (9 samples history as input).

The ELBND values were obtained from attributes of predictive model (prediction error, increment of adaptive weights) for every EEG electrode for every patient.

Two statistical functions applied on estimated ND were used to created criteria to decide whether the EEG records belong to healthy person of patient with dementia. First investigated function was standard deviation and the second one was entropy. For every function a different length of history as an input of predictor was used – 4 samples back for standard deviation extraction, 9 samples back for entropy extraction.

Every patient from the data-set has a multiple EEG records. From every record, it was used data recorded by electrodes 13 to 19. Records has different length, so the records were segmented into 1000 samples chunks (7.8125 seconds). The ELBND coefficients of segment were estimated in third epoch of LNU training. From the ELBND output of every segment are estimated standard deviation and entropy. So multiple values are obtained for one patient (depends on lengths of patient records). For classification of every patient, the average of those values was used - one average value for standard deviation and one average value for entropy.

For validation of the method, non-exhaustive cross-validation was used. We split patients between two groups (2-fold cross-validation). One group was for training (setting the criteria) and the other one for testing. Every group contains the same amount of patients with and without dementia. To eliminate the error caused by splitting into the groups, the patients were split into groups randomly 100 times for every tested criteria. The average of all results was estimated and presented as a final result. Three different criteria were used for patient classification. The first one utilizes just standard deviation of ELBND, second works just with entropy of ELBND, and the last one uses both functions.

The criteria based on standard deviation was just median of all patients from the training group. Patients from the testing group with lower value than median were classified as healthy ones and those with higher value as patients with dementia. This criteria gave as both specificity and sensitivity over 88%. The distribution of values of this criteria is in Fig. 5.5.

The entropy based criteria was build on the finding that demented patients have much bigger dispersion of novelty detection entropy than healthy patients (that is obvious from Fig. 5.5). So from the training group the lowest and highest value

for normal patients and the lowest and highest value of demented patients were estimated. Patients below lowest value of normal or above highest value of normal were considered as demented. That means that all demented patients with entropy in range of entropy dispersion of normal patients were marked incorrectly. That is reason why this criteria has much lower sensitivity and specificity than first (standard deviation based) criteria. The sensitivity of this criteria is 82% and the specificity is 66%.

The last criteria uses both statistical functions. The main part of this criteria is standard deviation. This value was further modified with a penalization for entropy. If the patient has entropy in normal range of testing group, the penalization is 0. If the entropy is below the normal range, there is linear penalization according to formula

$$P_i = \frac{en_{normL} - en_i}{en_{normL} - en_{dementL}} \cdot C, \tag{5.1}$$

where C stands for criteria, P_i stands for penalization of i-th patient, en_{normL} is lowest value of normal patients from training group, $en_{dementL}$ is lowest value of demented patients from training group and en_i is value for i-th patient. This criteria has best classification results. The sensitivity and specificity are both 90%. Dispersion is shown in Fig. 5.5.

The novelty of EEG signal of 110 normal and 110 demented patients was estimated. Three different criteria for classification of patients were used. The best result was obtained with criteria that uses features of both other criteria. With the method that was proposed in this study and this model settings that was used, the specificity and sensitivity of 90% was achieved. Results of all criteria are summarized in Table 5.1. The results of some of other methods are in Table 5.2.

Criteria based on	Specificity	Sensitivity
Standard deviation	88%	88%
Entropy	65%	82%
Standard deviation and entropy	90%	90%

Table 5.1: Table of results for classification based on ELBND method

Method	Specificity	Sensitivity
Fractal Dimension Measure	99,9%	67.00%
Probability Density Function of the Zero-crossing Intervals	99,9%	78.00%
Approximate Entropy at P3	100.00%	70.00%
Approximate Entropy at P4	75.00%	80.00%
Other studies of American Academy of Neurology	70.00%	81.00%

Table 5.2: Table of results for other methods

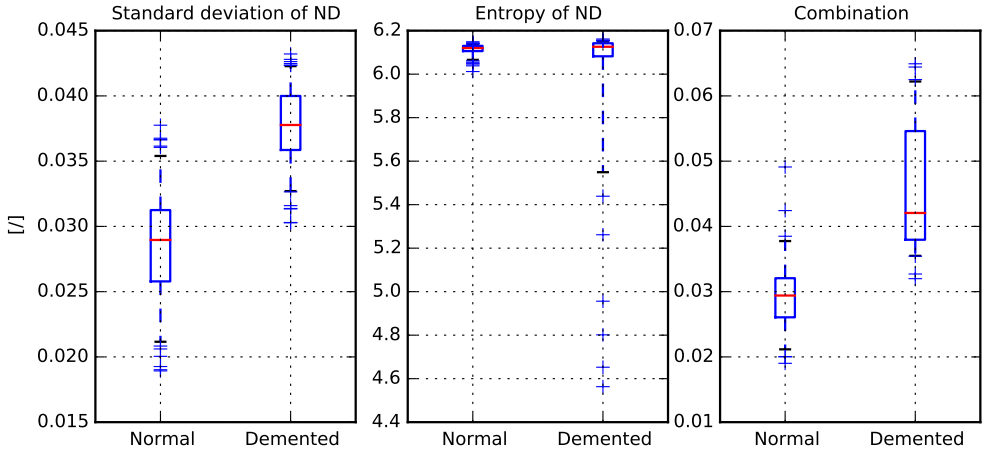

Figure 5.5: Box and whisker plots of results for all tested classification criteria

5.2 Dealing with concept drift

The potentials of the proposed ELBND method for novelty detection with drifted real-time data are presented in this section. This potentials were tested in two studies [mc9] and [mc2]. The explanation of the concept drift and its modeling is in subsection 5.2.1. The framework used for testing and cross-validation of both related studies is described in subsection 5.2.2. The description of reference signals and methods used for comparison in the studies is in subsection 5.2.3. The results of the studies can be found in subsection 5.2.4. Finally the conclusion of the studies is in subsection 5.2.5.

5.2.1 Modeling of concept drift

Most of methods in machine learning field is mainly focused on learning from data assumed to be drawn from a particular distribution [76]. However, the modern industry brings a new challenges like highly non-stationary data obtained in real-time as data streams [77]. Dealing with these data streams has multiple difficulties. One of them is impossibility to remove noise with advanced methods that use knowledge of future samples. Other difficulty is concept drift [78]. The concept drift is well known name to describe that the statistical attributes of the observed variable change over time in unforeseen ways. The concept drift is a cause of a significant problems for all methods that rely on data long term statistic attributes (thresholding, etc.).

In a field of novelty detection, the concept drift is considered as a challenging data imbalance that should be ignored, and only system changes and outliers that represent novelty should by highlighted by the novelty detection methods. In other words, the drift in general does not represent a novelty. The field of application for such novelty detection methods is broad. For example, the method can be used as a supportive method for real-time system fault detection, for onset detection of events in biomedical signals, monitoring of non-linearly controlled processes, or event driven automated trading, etc..

As it was mentioned allready in the introduction chapter, the intuitive way how to deal with concept drift is adaptation [79, 78, 80]. The learning model adaptation can compensate the drift gradually and thus makes the adaptive model a suitable novelty detector. Such a novelty detector can successfully ignore the concept drift that is not considered to be a novel event. Although novelty detection via learning of adaptive models is a topic already researched for a few decades [13, 10], the most of the published methods are limited rather for a specific purpose. At least, there are not tested for more general use cases.

It is important to highlight that most of the developed novelty detection methods works as independent systems. However, in the age of Big Data, when every possible information is logged in the most raw form, the adaptive methods can be used on already implemented devices and solutions (predictors, filters, controllers).

The adaptive models such as adaptive filters, fuzzy systems and neural networks become an essential technologies in a great array of technological processes. Thus the adaptive novelty detection is another option how to use already implemented tools to optimize and improve processes for minimal computational cost.

No unified theory how to categorize concept drift exists. This is because the categorization of different types of concept drift is a complex task [80]. However, various categorizations exists according to purpose of the categorization. The most simple categorization has only two categories of concept drift: abrupt and gradual [81]. However, others [82] reference the gradual drift as concept drift and the abrupt change as concept shift. The examples of gradual, recurring and abrupt drift are shown in Figure 5.6. For complex categorization of concept drift see [80]. Another categorization [83] is separating concept drift into these two types of concept drift: real (concept shift) and virtual (drift that does not influence target concept [84]).

In case of simple adaptive models like adaptive filter, the concept shift (abrupt change) can be hard to ignore. Such a sharp change always excites adaptation and thus is emphasized by attached adaptive novelty detector. In general such a sharp changes are not that difficult to detect. For this reason the study focuses only on most basic type of concept drift - gradual drift.

Two types of gradual concept drift were simulated in the studies presented in the next subsections. Two models of concept drift were used in mentioned works:

- Ramp (pure gradual drift - slow constant increment). This kind of drift is problematic because greater distance from data zero mean can alter the adaptive model performance and also it makes any standardization (z-score) impossible.

- Harmonic wave (periodically repeating drift). This drift is also known as recurring trend. This drift is commonly present in various biological systems and stock market behavior.

These two models cover the most of the features of concept drift in time series. The constant increment drift was starting from zero and finishing in 1 at the end of simulation. The sinus like drift has period of 10^4 samples and amplitude of 1.

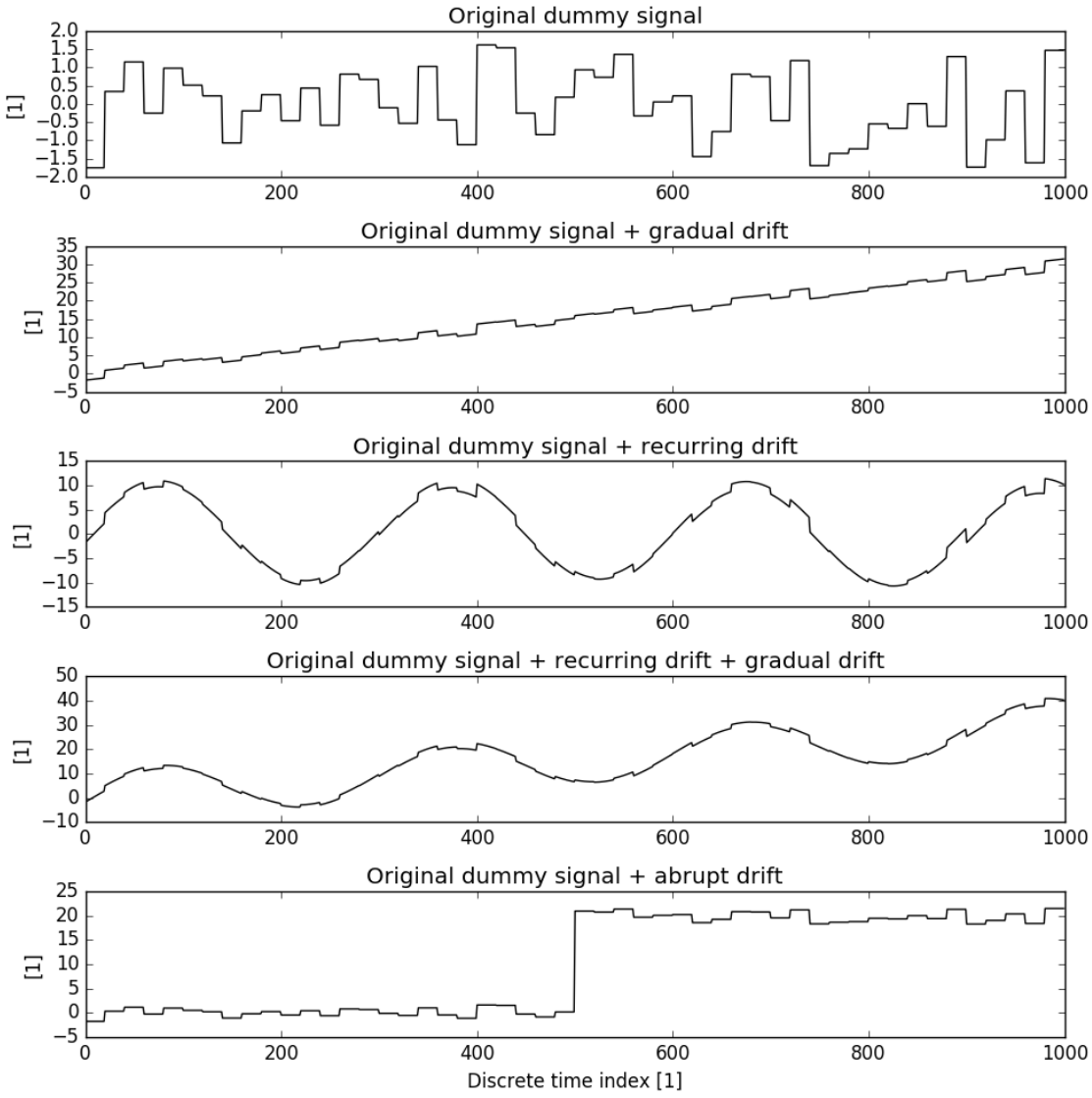

Figure 5.6: Examples of concept drift effect on synthetic dummy signal.

5.2.2 Testing framework and cross-validation

The ELBND method is used as an unsupervised feature extractor implemented on supervised learning model. The feature extraction is a process that derives values (features) intended to be informative and non-redundant. Because of this reason, any direct empirical comparison of the proposed method can be done only through the performance of a classifier that uses the extracted features. A suitable classification framework is used for this task and it is described in this subsection.

The ELBND (and the reference method LE) have already demonstrated their applicability on real life task in past [mc6, mc1, mc10] and [85]. In general, however,

the real tasks are too specific for objective comparison of multiple methods. This problem exists because of multiple reasons. The major problems of real data involve:

- In real measured data it is hard to find and adjust the level and type of the noise.
- The correct positions of a novel event can be unobtainable in larger scale.
- The exact positions of a novel event cannot be annotated with high precision.
- Furthermore, the process of annotation of a novel event onset is often opinion based.

Because of the reasons above, a synthetically created data that fully model the general challenges of novelty detection in a large scale was used. The concept drift was also modelled and added into the data. The inspiration for this solution comes from [86].

Both mentioned studies used the similar classification framework as in the older study [mc4]. The ELBND and all other methods in mentioned studies work as a feature extractor. Thus the classifier used for method comparison has only the purpose to detect weather the extracted feature (level of novelty) rises only on segments of data where the target event occurs (change point=system change, perturbation=outlier). In other words, the classification can be described (Fig. 5.7) as:

- true condition: 1 is in the position of novel event and some number of samples after, 0 is everywhere else. This position is annotated during the process of data creation.
- predicted condition: 1 for time index where the novelty rises over threshold, 0 everywhere else.

The example of classification process is displayed in Fig. 5.7. The present threshold is moving in full range to test significant number of settings to build a receiver operator characteristic (ROC) curve. The interpretation of possible output conditions is displayed in Tab. 5.3.

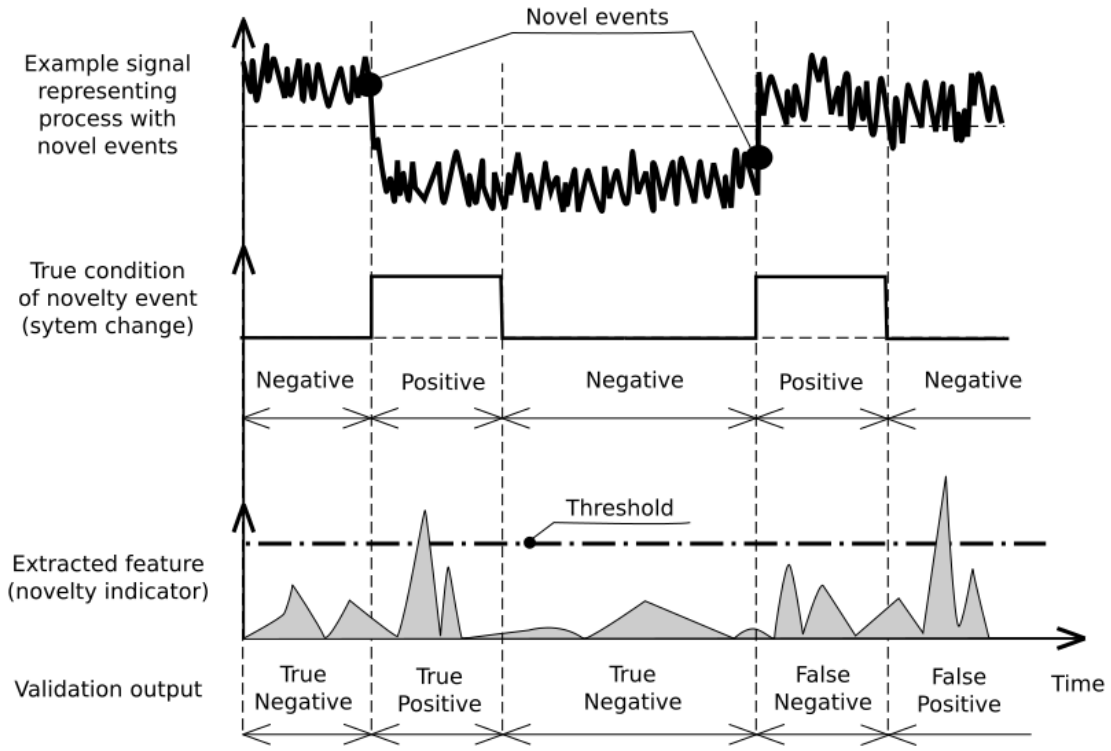

Figure 5.7: The very principle of the classification framework used for cross-validation of the two new novelty detection methods (ELBND, LE) and two bench-marking ones (plain error, SE), where the true conditions are the desirable objectives (bottom axes). The interpretation of classifier output is explained in Tab. 5.3 in more detail.

The area considered as surroundings of the novel event is some number of samples after the novel event occurrence. This number is trade-off based on two requirements:

1. An adaptive model needs some time to adapt to a new (changed) process, thus the error and other parameters are high for a while after a novel event.

2. A novelty detection measure should react fast to a novelty and should not merge together two events close in time.

Note that this cross-validation setup results in the same number of positive segments as negative segments. In other words the classification data-set is balanced and its segments has the same length.

True condition	Detector output	
	Positive finding	**Negative finding**
Novel event (true positive)	True positive	False negative
No change (true negative)	False positive	True negative

Table 5.3: Interpretation of classifier possible output conditions (true condition = presence of novel event, finding = actual result from classifier).

5.2.3 Reference methods and signals

5.2.3.1 Error of prediction

The simplest reference signal is just the plain error of the adaptive model. This reference is interesting because it is the most easiest feature describing novelty in data that is possible to obtain from adaptive model. Usage of the error as a reference directly displays how much information can be obtained from the adaptive model with more sophisticated methods like ELBND or LE. That is the reason why it is used in study [mc9] for result validation.

5.2.3.2 Learning entropy (LE)

The learning entropy (LE) is more advanced but similar learning-based method to ELBND. That is the reason why it is useful to compare it with ELBND in this thesis. The method called the LE is also called the approximate individual sample learning entropy (see AISLE in [87]), however in this thesis and in some mentioned studies it is called with shorter name learning entropy and it is obtained as follows

$$\mathrm{LE}(k) = \frac{1}{n \cdot n_\alpha} \sum f(\Delta w_i(k), \alpha) \ ; \ \forall \alpha \in \boldsymbol{\alpha} \ , \tag{5.2}$$

where n is the number of adaptive weights and n_α is the number of user defined detection sensitivities

$$\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_{n_\alpha}] \ ; \ \alpha_1 < \alpha_2 < \ldots < \alpha_{n_\alpha}. \tag{5.3}$$

The function $f(\Delta w_i(k), \alpha)$ is defined as follows

$$f(\Delta w_i(k), \alpha) = \begin{cases} 1, & \text{if } |\Delta w_i(k)| > \alpha \cdot \overline{|\Delta w_{Mi}(k)|} \\ 0, & \text{otherwise} \end{cases} \tag{5.4}$$

where $\overline{|\Delta w_{Mi}(k)|}$ is the mean value of the window used for LE evaluation and depends on the data and possible periodicity [87]. Also the optimal number of detection sensitivities and their values is optional, and it should be chosen within the range where the function LE(k) returns a value lower than 1 for at least one sample in the data, and at most for one sample returns value of 0 on pre-training data [87].

5.2.3.3 Sample Entropy

Sample entropy (SE) is a modification of approximate entropy (ApEn), used for assessing the complexity of time-series signals, mainly used for physiological time-series and diagnosing diseased states [88]. The SE is a proven to be a conventional tool for detection of novelty events. It was studied for detection of epilepsy in clinical applications [89]. In [90], neonatal sepsis detection from abnormal heart rate characteristics was studied.

The SE was chosen as a benchmark tool for the study [mc9] because it annotates the samples in similar way like the proposed method ELBND, or the similar adaptive method LE. That is the reason why it is interesting for direct comparison also in this thesis.

5.2.4 Experiments and results

Data for all simulation were created synthetically to achieve uniform occurrence of novel events among data. The detailed reasons for this solution were explained in introduction. Two different novelty detection tasks were created for the experimental analysis. First task is the system change point detection (contextual novelty detection) with a system that can be completely modeled by a used adaptive model. Second task is the outlier detection (value based novelty detection) in a more complex signal (ECG waveform) that is not possible to fully model with a given adaptive

model. These two tasks were selected because every one of them represents different challenge for adaptive models and novelty detectors. The experiments are explained in detail in following subsections. All simulations have been done in language Python.

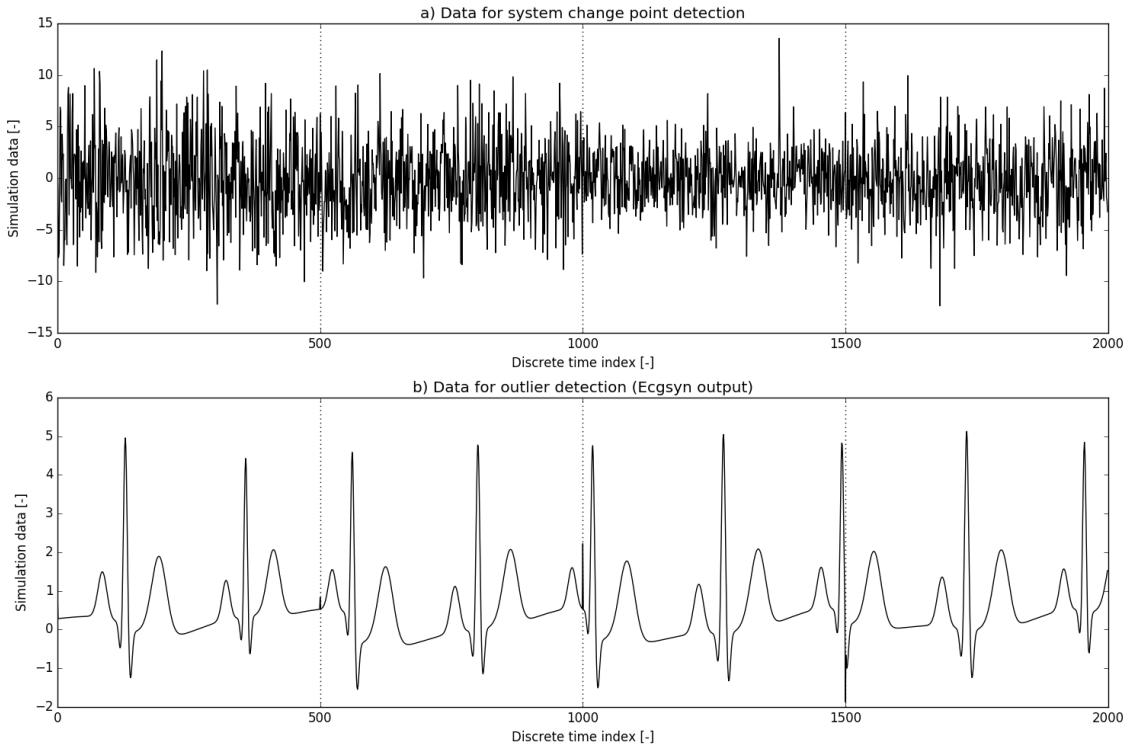

Figure 5.8: Data used for detection and validation (each of 250 000 samples in total). Doted vertical lines mark the positions of novelty occurrences (of random magnitudes): a) the detail of first data set is the output of system with system changes as novelty; b) the part of second data set for outlier detection - Ecgsyn generated ECG (waveform with perturbations - outliers)

5.2.4.1 System change point detection with NLMS

This experiment can be found in study [mc9]. The goal of this experiment was to test the ability of ELBND, LE and SE to detect novel events (system change point) in data. The results of the experimental analysis is interesting for this thesis, because it provides some new information on ELBND novelty detection potentials. Especially it provides comparison between adaptive based methods (ELBND, LE and error of prediction) and the conventional method SE.

Drift	Method	Maximal accuracy [%]	AUROC[%]
none	LE	88.687	95.262
none	ELBND	91.01	96.295
none	ERR	89.394	95.519
none	SE	50.808	48.854
ramp	LE	76.162	81.347
ramp	ELBND	71.818	80.276
ramp	ERR	71.515	79.579
ramp	SE	50.505	48.714
sinus	LE	69.596	75.474
sinus	ELBND	67.374	74.602
sinus	ERR	65.657	72.502
sinus	SE	51.515	48.723
both	LE	70.303	75.039
both	ELBND	65.96	68.266
both	ERR	64.747	67.906
both	SE	50.909	49.563

Table 5.4: Table of results for system change point detection (change point is the novel event). The process with novel events is represented by equation 5.5.

In order to achieve a general data-set for testing novelty detection, first, a system that is possible to be modeled by an adaptive filter with zero error was created. In other words, the model should be able to recognize novel event in data in all cases without a mistake. This ideal system was contaminated with noise and concept drift (5.5) to make the task of novelty detection difficult. With this setup it was possible to monitor how difficult is the environment for the adaptive model.

The used data $y(k)$ were generated according to the following equation

$$y(k) = h_1(k)x_1(k) + ... + h_n(k)x_n(k) + \xi(k) + \chi(k), \tag{5.5}$$

where $h_i(k)$ are parameters of the data generator, $x_i(k)$ are input variables, $\xi(k)$ is noise and $\chi(k)$ represents the concept drift. Ten independent series of white

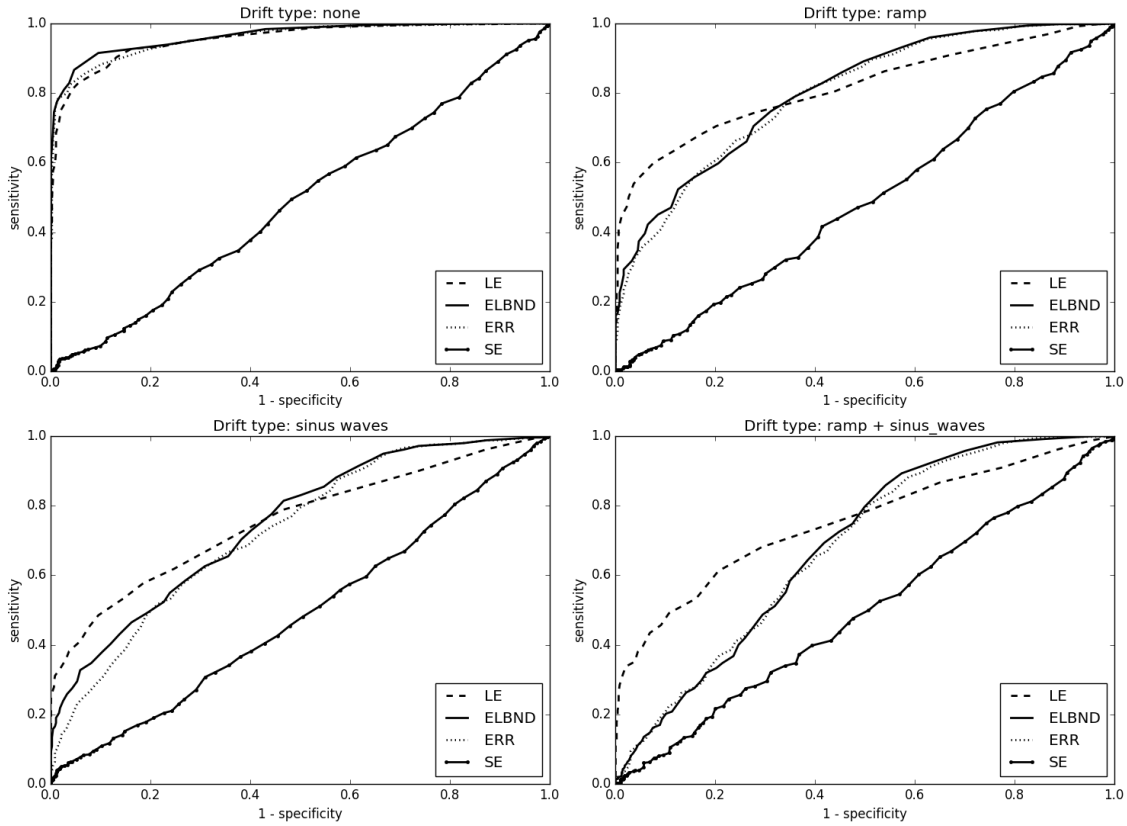

Figure 5.9: The ROC curves for system change point detection analysis (ERR - error of prediction, SE - based on sample entropy).

Gaussian noise with unit standard deviation and zero mean were used as the input. The generator parameters $h_i(k)$ change randomly every $n_{change} = 500$ samples. These changes of parameters are sharp and with unit standard deviation and zero mean. The data contains 500 of such changes. The example of the resulting data can be seen at Figure 5.8a.

The signal to noise ratio (SNR) was evaluated with following formula

$$\mathrm{SNR} = 10 \log_{10} \frac{\sigma_y^2}{\sigma_\xi^2}, \tag{5.6}$$

where σ_y is a standard deviation of unknown system output and σ_ξ is a standard deviation of noise $\xi(k)$. The level of noise for this simulation was 10.429dB on average. Note that the level of noise was slightly different in every segment of data. This variation is caused by different parameters $h_i(k)$ of generator.

The adaptive filter was used in predictive settings with $n = 10$ adaptive parameters (a parameter for an input). At the beginning, the parameters were set to zeros.

Drift	Method	Maximal accuracy [%]	AUROC[%]
none	LE	90.909	94.127
none	ELBND	87.273	88.094
none	ERR	81.111	82.661
none	SE	51.717	50.023
ramp	LE	73.131	76.698
ramp	ELBND	59.596	62.749
ramp	ERR	55.657	56.112
ramp	SE	50.505	48.39
sinus	LE	70.404	73.491
sinus	ELBND	60.505	64.107
sinus	ERR	56.162	57.617
sinus	SE	52.828	52.308
both	LE	72.121	76.137
both	ELBND	57.374	60.645
both	ERR	54.646	55.283
both	SE	52.424	51.72

Table 5.5: Table of results for outlier detection (the occurrence of an outlier is the novel events). The outliers are the perturbations in Ecgsyn output (Fig.2b).

Initial value for adaptive learning rate was set to $\eta(k) = 1.5$.

The results of the experiment were evaluated by three different metrics: AUROC, maximal accuracy (MAX ACC) and ROC (for graphical comparison). More detailed information about these used cross-validation tools can be found in section 2.2.

The resulting receiver operator curve (ROC) are shown in Fig. 5.9. Maximal accuracy and area under the receiver operator curve (AUROC) are displayed in Tab. 5.4.

5.2.4.2 Outlier detection with NLMS

The goal of this analysis was to correctly detect occurrence of perturbations in data. Synthetic electrocardiography (ECG) data was used for this study. To generate syn-

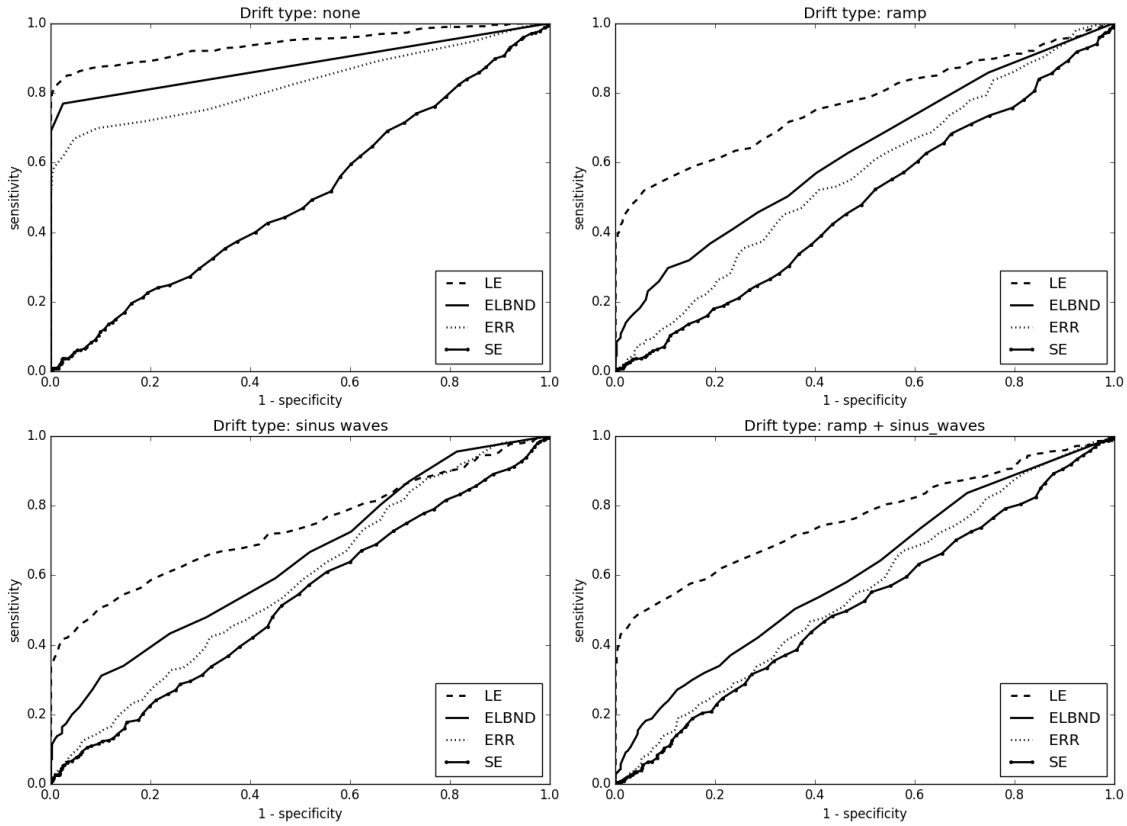

Figure 5.10: The ROC curves for outlier detection analysis (ERR - novelty detection based on error of adaptive model, data Fig.2b).

thetic ECG data, the Ecgsyn [91] (a realistic ECG waveform generator) was used. Parameters of the generator were set as follows: sampling frequency was 256Hz, mean heart rate was 60 beats per minute, standard deviation of heart rate was 1 beat per minute, LF/HF ratio was 0.5, internal sampling frequency was 512Hz, angles of PQRST extrema was set to $[70, -15, 0, 15, 100]$, z-position of PQRST extrema was set to $1.2, -5, 30, -7.5, 0.75$ and Gaussian width of peaks was set to $[0.25, 0.1, 0.1, 0.1, 0.4]$.

Perturbations (simulated outliers) were introduced into this simulated waveform time-series. The outliers were random numbers from normal distribution with zero mean and 0.1 standard deviation (standard deviation of non drifted ECG signal is 0.907). These outliers were added to the signal values. One outlier was placed at every 500 samples. Total number of introduced outliers was 500. The example of the resulting data can be seen at Figure 5.8b.

The resulting receiver operator curves (ROC) are shown in Fig. 5.10. The maximal accuracy and the area under the receiver operator curve (AUROC) are displayed in Tab. 5.5.

The adaptive filter was used in predictive settings with $n = 10$ adaptive parameters (a parameter for an input). At the beginning, the parameters were set to zeros. Initial value for adaptive learning rate was set to $\eta(k) = 1.5$. Note that the average period of one ECG wave is about 25.6 times greater than history used for prediction ($n = 10$). Thus, in this experiment the adaptive model cannot fully learn the dynamic behind the ECG generating process. In other words, the model will always have some prediction error, no matter how long it will learn.

5.2.4.3 Comparison of system change point detection with NLMS, NLMF, RLS and GNGD

The results presented here are obtained from study [mc2]. Simulated data were used in this study to compare the performance of ELBN and LE in dependency of used adaptive filter (NLMS, NLMF, RLS, GNGD). The main contribution of this study for this thesis is the performance overview of ELBND used with various adaptive filters.

The data used in the study were generated according to the following equation

$$y(k) = h_1(k)x_1(k) + ... + h_n(k)x_n(k), \tag{5.7}$$

where $h_i(k)$ are parameters of the process and $x_i(k)$ represents input variables. There were ten input variables. Each one of them was independent series of white Gaussian noise with unit standard deviation and zero mean value. The process parameters $h_i(k)$ were randomly changed every 500 samples. These sharp changes were introduced to the data as the target novelty. The actual values of the process parameters $h_i(k)$ were taken from normal distribution with standard deviation of 0.5 and zero mean value. The data-set contain 500 of such sharp changes representing novel events (250×10^3 samples). The generated data were contaminated with *additive white Gaussian noise* (AWGN). At the end, the concept drift was added to the data. The concept drift was modeled as a harmonic wave with period of 10^4 samples and

varying amplitude. First two waves (20×10^3 samples) were used for training of adaptive filters. The rest of the data was used for testing.

The process of data generation described above is also shown graphically in Figure 5.11 (drift period=10000, drift amplitude=5, SNR = 10db).

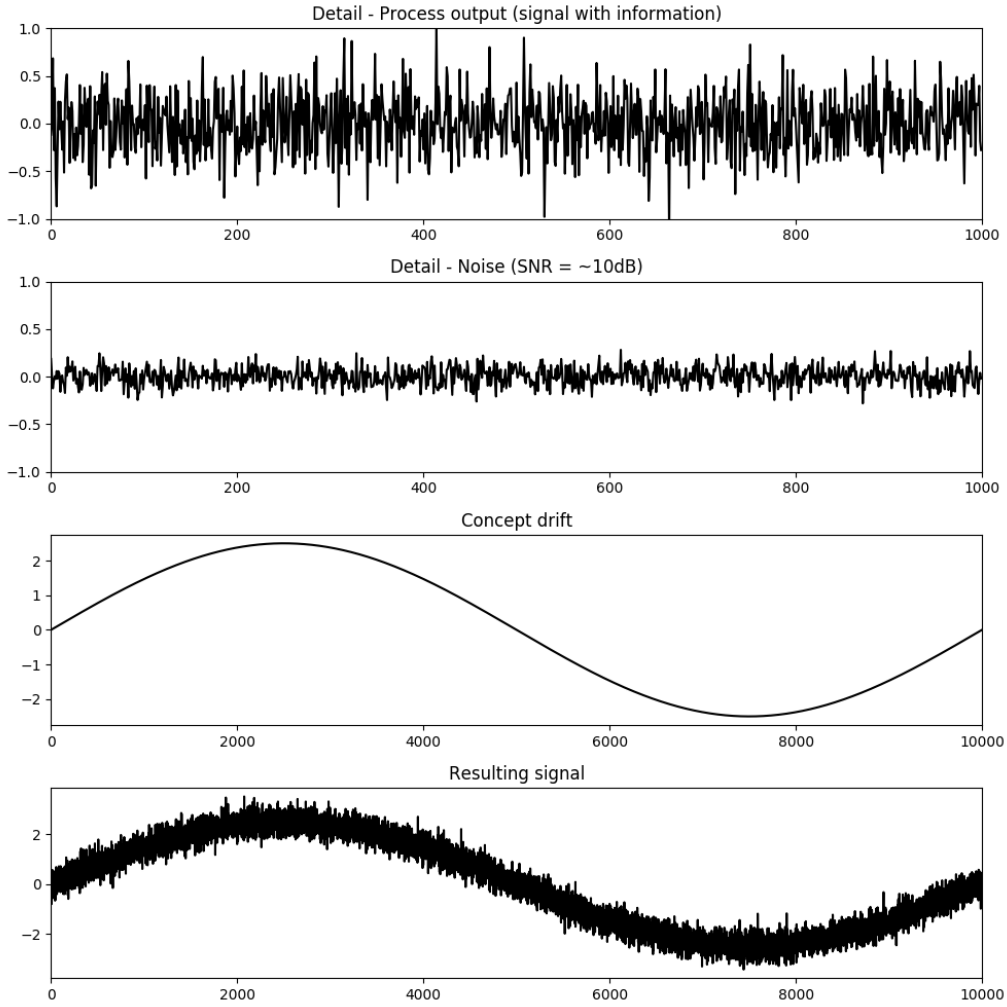

Figure 5.11: The simulated signal representing output of simulated system and its components.

The results of the experiments were evaluated by three different metrics: AUROC, maximal accuracy (MAX ACC) and ROC (for graphical comparison). More detail information about these used cross-validation tools can be found in section 2.2.

The graphical results - the ROC curves are shown in figures: Figure 5.12, Figure 5.13, Figure 5.14, Figure 5.15, Figure 5.16 and Figure 5.17. The AUROC and maximal accuracy results are in Table 5.6.

Adaptation	Detection	AUROC [%]	MAX ACC [%]
SNR = 5.4dB; drift amplitude = 5 [-]			
NLMF	LE	73.9	68.6
	ELBND	83.7	75.7
NLMS	LE	53.1	53.9
	ELBND	53.6	53.5
GNGD	LE	63.6	58.9
	ELBND	65.8	61.7
RLS	LE	79.6	71.9
	ELBND	87.6	81.5
SNR = 5.4dB; drift amplitude = 2 [-]			
NLMF	LE	81.0	74.4
	ELBND	96.1	90.3
NLMS	LE	57.9	56.7
	ELBND	63.1	60.2
GNGD	LE	74.2	68.1
	ELBND	77.9	71.3
RLS	LE	88.3	80.1
	ELBND	96.4	90.6
SNR = 5.5dB; drift amplitude = 0 [-]			
NLMF	LE	82.3	76.2
	ELBND	96.7	91.5
NLMS	LE	61.5	60.5
	ELBND	71.3	65.3
GNGD	LE	79.1	73.1
	ELBND	83.8	77.3
RLS	LE	91.5	83.3
	ELBND	96.8	91.4

Adaptation	Detection	AUROC [%]	MAX ACC [%]
SNR = 24.0dB; drift amplitude = 5 [-]			
NLMF	LE	93.4	87.6
	ELBND	99.0	96.9
NLMS	LE	65.5	62.0
	ELBND	56.2	58.6
GNGD	LE	81.9	74.1
	ELBND	72.0	65.5
RLS	LE	90.3	83.1
	ELBND	95.0	89.5
SNR = 24.1dB; drift amplitude = 2 [-]			
NLMF	LE	97.4	94.0
	ELBND	99.5	97.9
NLMS	LE	85.8	80.2
	ELBND	84.7	75.5
GNGD	LE	94.8	88.3
	ELBND	94.9	88.0
RLS	LE	97.7	92.4
	ELBND	99.3	96.0
SNR = 24.0dB; drift amplitude = 0 [-]			
NLMF	LE	96.1	91.9
	ELBND	99.0	97.1
NLMS	LE	99.9	99.9
	ELBND	100.0	100.0
GNGD	LE	100.0	99.9
	ELBND	100.0	100.0
RLS	LE	100.0	99.7
	ELBND	99.6	97.9

Table 5.6: The results of the ELBND and LE comparison with various adaptive filters. Results for experiments with high level of noise are on the left side, the results for experiments with the level of noise are on the right side.

According to the obtained results, it seems that ELBND has better potential with NLMF and RLS algorithms, while the LE works better with GNGD and NLMS algorithms. It is not surprising because the GNGD and NLMS are very similar gradient methods.

5.2.5 Summary

In this section all results related to the ELBND method used for data with concept drift were presented . Multiple novelty detection tasks were simulated to test the method and their suitability for system change detection and for outlier detection under the occurrence of concept drift, which usually complicates the detection as it appears from comparison to merely detection via plain error based detection and sample entropy based detection.

The results from both presented studies displays that ELBND can compete to

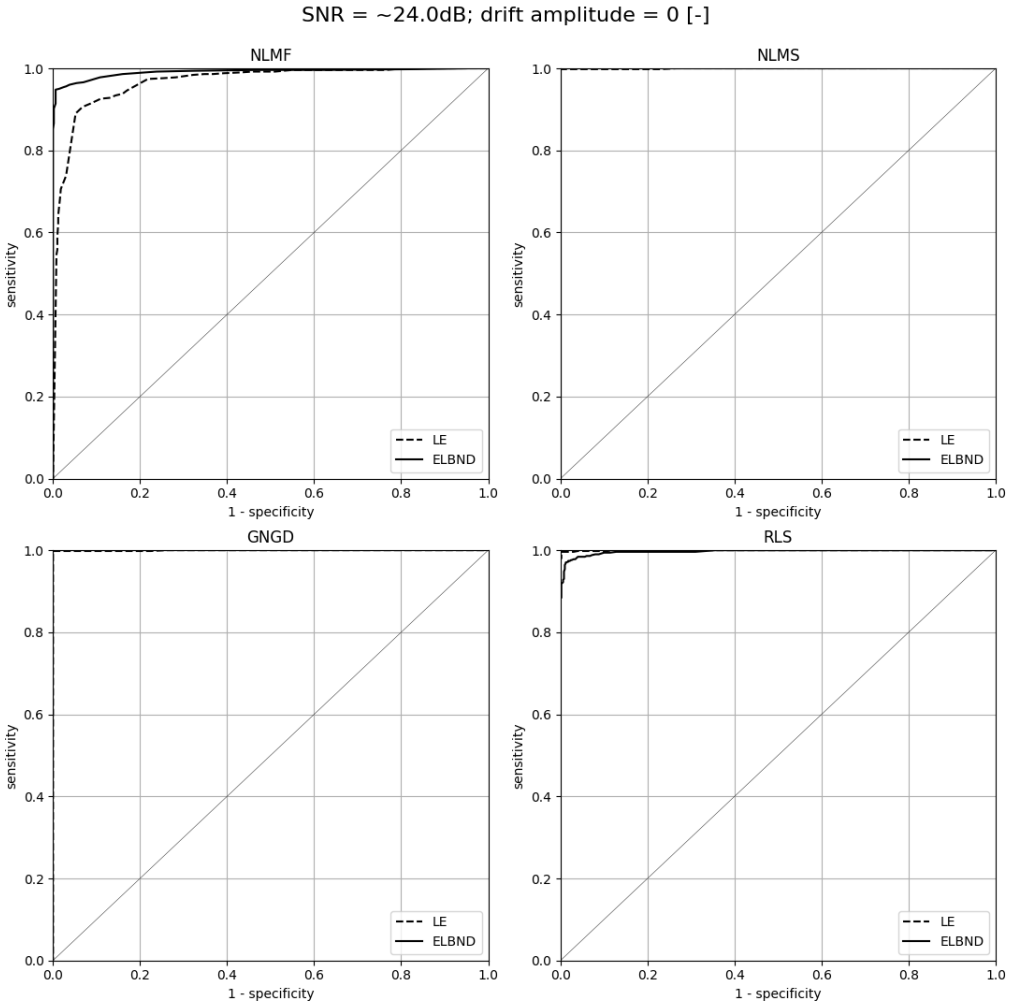

Figure 5.12: ROC curves for data without drift and with low level of noise (SNR = 24.0dB). Empty plots represents zero or almost zero detection error. The plot is adopted from study [mc2].

LE in various cases. Furthermore, the ELBND extracted feature is generally better than just plain prediction error for novelty detection. According to the second study, it seems that the LE works especially well with RLS and NLMF algorithm. The most important finding is, that the ELBND works in all cases better than the SE algorithm.

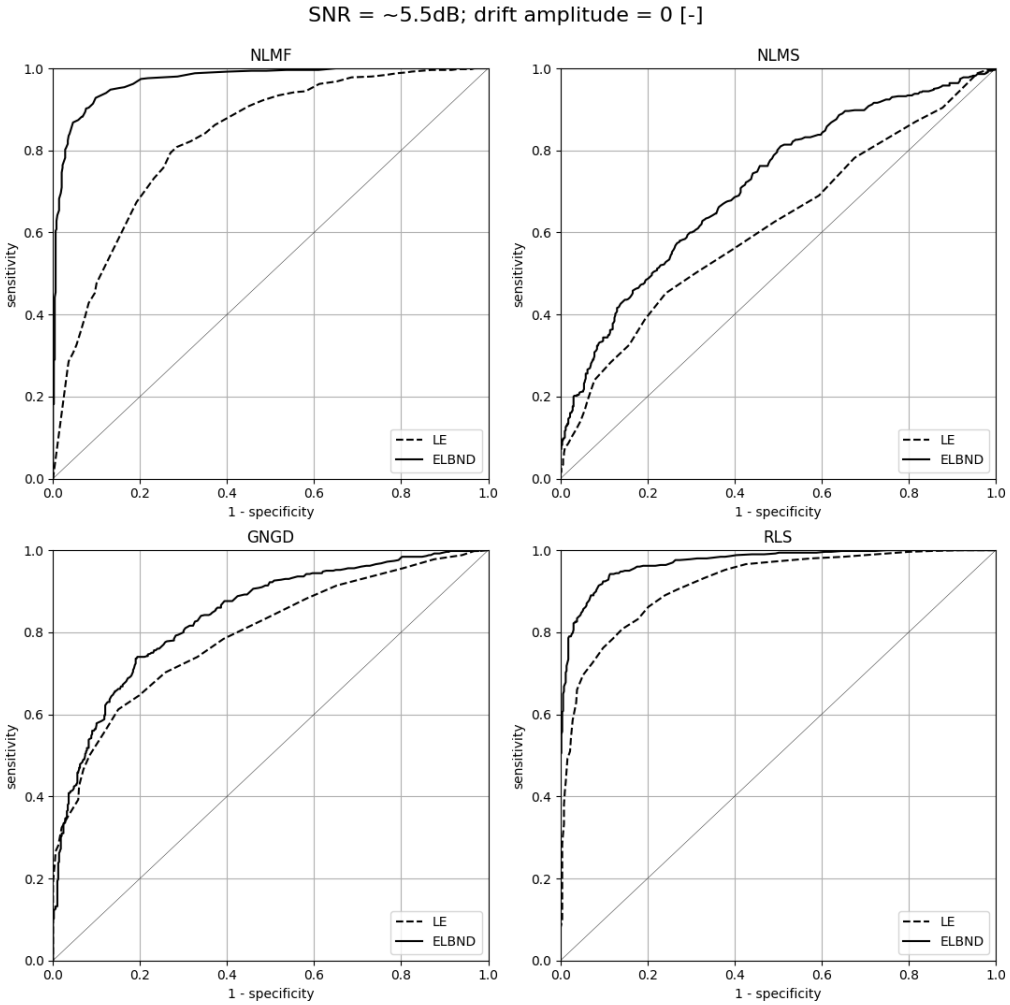

Figure 5.13: ROC curves for data without drift and with high level of noise (SNR = 5.5dB). In this case the ELBND yields better results than LE for all tested adaptive filters. The plot is adopted from study [mc2].

5.3 Other experiments

In this section results from studies that does not fit in previous sections are presented. In subsection 5.3.1, it is presented that ELBND has its potential for system change point detection. The investigation of noise level influence on ELBND performance is presented in subsection 5.3.2. The study related to time complexity of the ELBND method is in subsection 5.3.3.

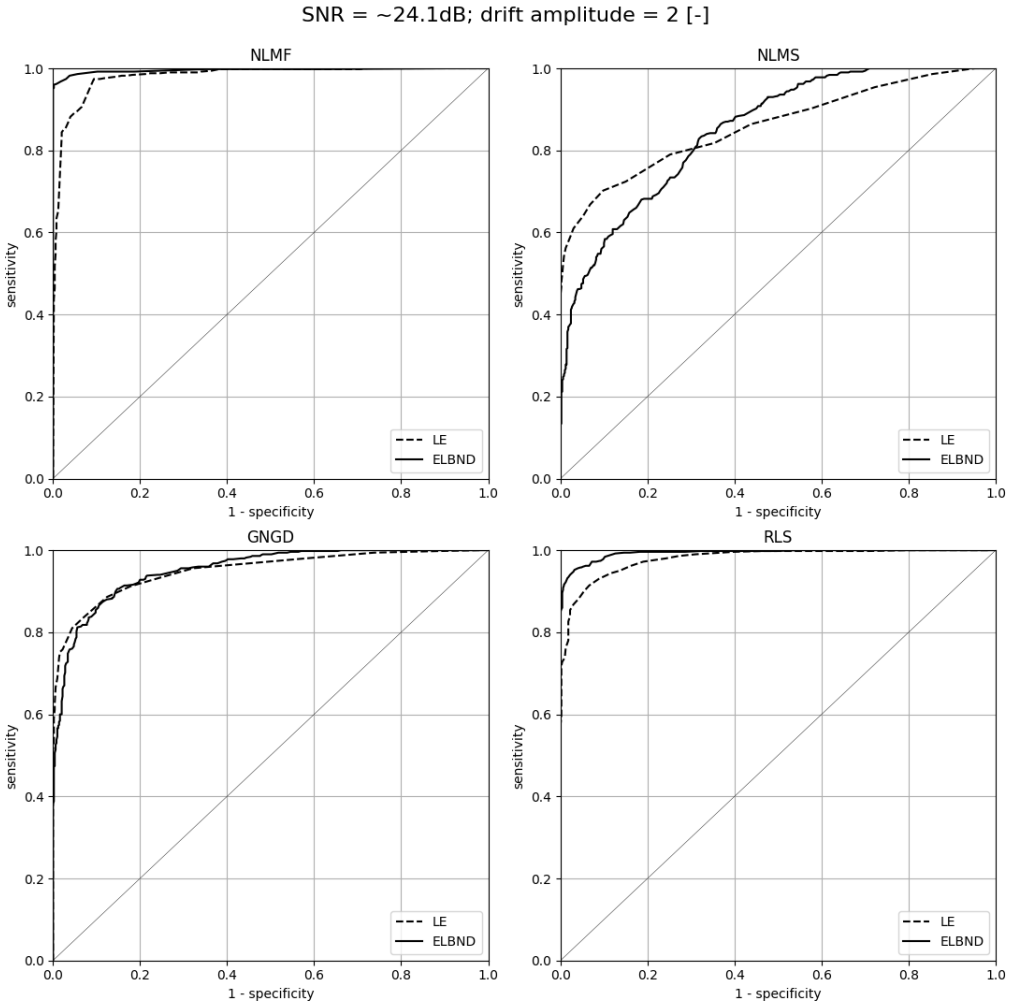

Figure 5.14: ROC curves for data with concept drift (drift amplitude=2) and with low level of noise (SNR = 24.1dB). In this case ELBND produces better results only for some adaptive filters. The plot is adopted from study [mc2].

5.3.1 System change point detection

This subsection is based on the results presented in [mc3]. For demonstration of the proposed method with LMS adaptation, a linear adaptive model (linear neural unit) was used. In this study, the novelty detection is used for detection of changes in a system. The change of a system is the plant model sensitivity and the model time constant. The simulated data with record of changed parameters are shown in Figure 5.18.

The results ared displayed in Figure 5.19. The squared error of the prediction or the presented novelty detection method produce visible high coefficients for samples

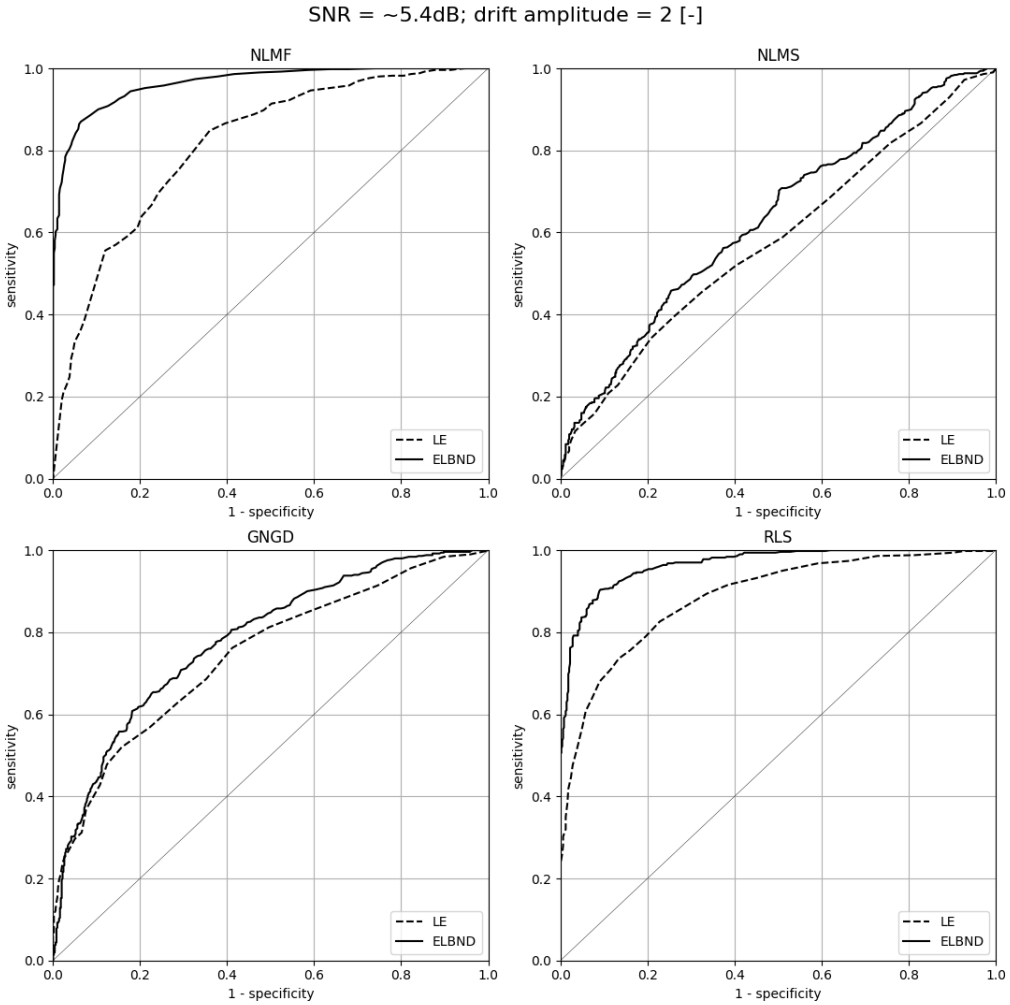

Figure 5.15: ROC curves for data with concept drift (drift amplitude=2) and with high level of noise (SNR = 5.4dB). In this case ELBND produces better results than LE with all tested adaptive filters. The plot is adopted from study [mc2].

where the system changes occur. So the coefficients could be used for visual or even automated detection of system changes (for example with implemented threshold).

The second tested method was RLS adaptation. The same data was used also for LMS based Novelty detection approach. The results of the RLS novelty detection example are displayed in Figure 5.20. In this results it is possible to see, that introduced technique could produce different and better information about novelty in data than just the squared error of prediction e^2.

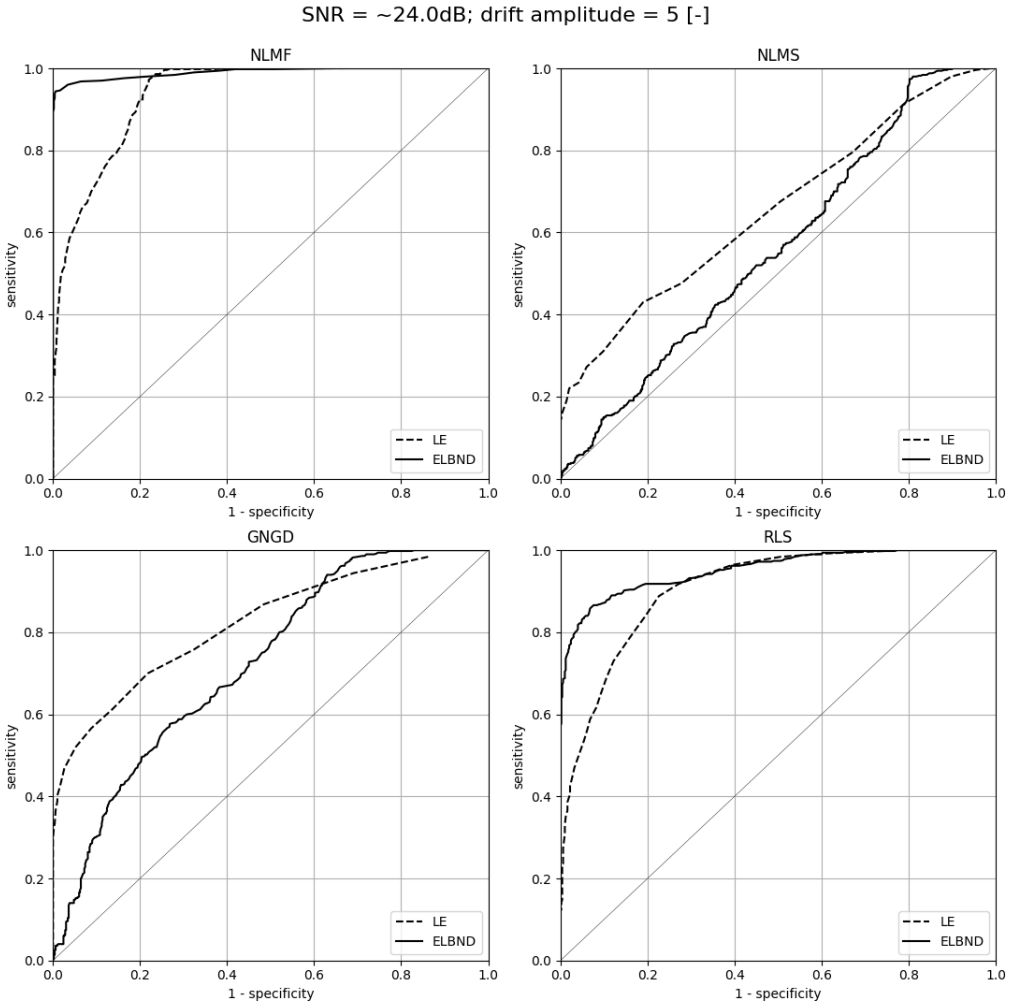

Figure 5.16: ROC curves for data with drift (drift amplitude=5) and with low level of noise (SNR = 24.0dB). In this case ELBND produces better results than LE only for some adaptive filters. The plot is adopted from study [mc2].

5.3.2 Influence of noise type and level on ELBND performance

This subsection is based on study [mc4]. The study investigates how the type and the level of additive noise contained in data influence the outcome of the ELBND and LE method.

The data-set for this study was created synthetically by simulation. This is due to the fact that for a real measured data it is difficult to set the noise exactly to the desired level in some consistent way. Also the exact positions of a process changes

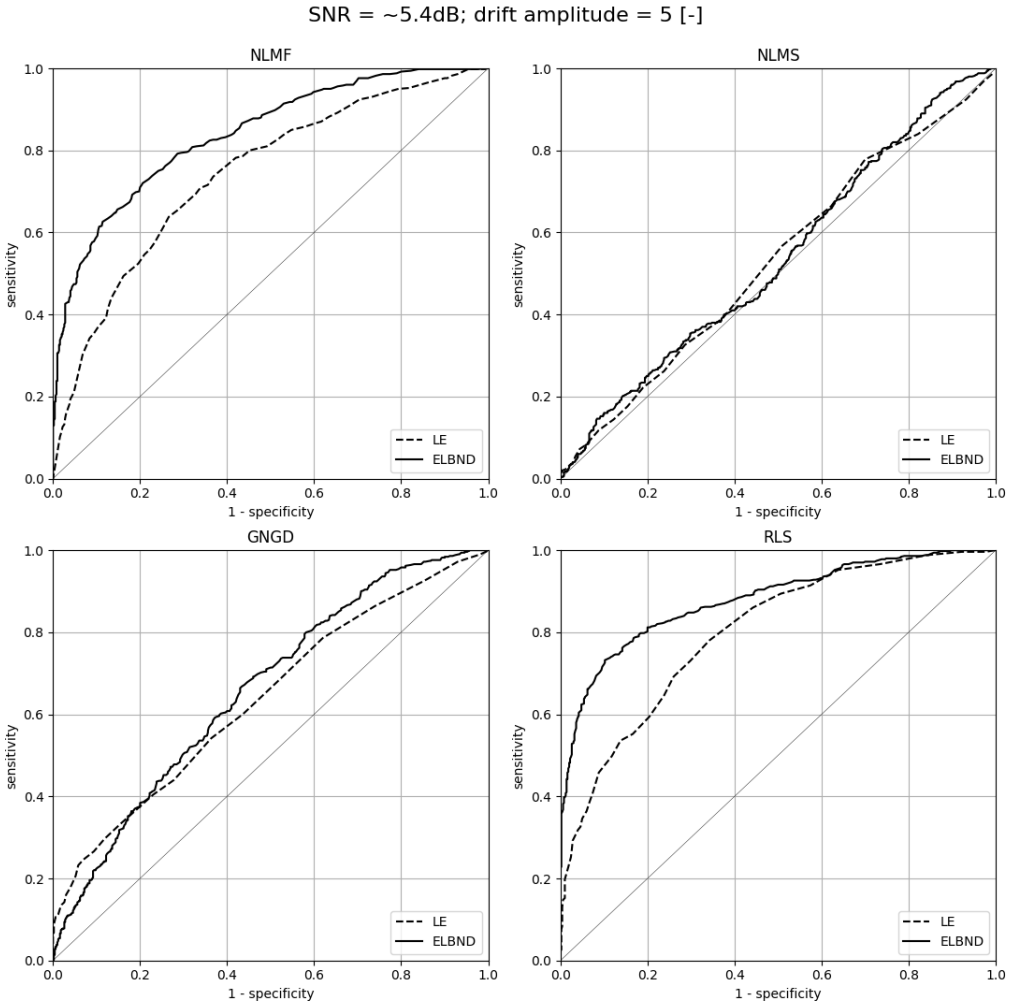

Figure 5.17: ROC curves for data with drift (drift amplitude=5) and with high level of noise (SNR = 5.4dB). In this case ELBND produces better results than LE only for some adaptive filters. The plot is adopted from study [mc2].

can be unobtainable at larger scale.

5.3.2.1 Experiment design

The used data $y(k)$ were generated according to the following equation

$$y(k) = h_1(k)x_1(k) + ... + h_n(k)x_n(k), \tag{5.8}$$

where $h_i(k)$ are parameters of the data generator and $x_i(k)$ are input variables. Ten independent series of white Gaussian noise with unit standard deviation and zero mean were used as the input of system 5.8. The generator parameters $h_i(k)$

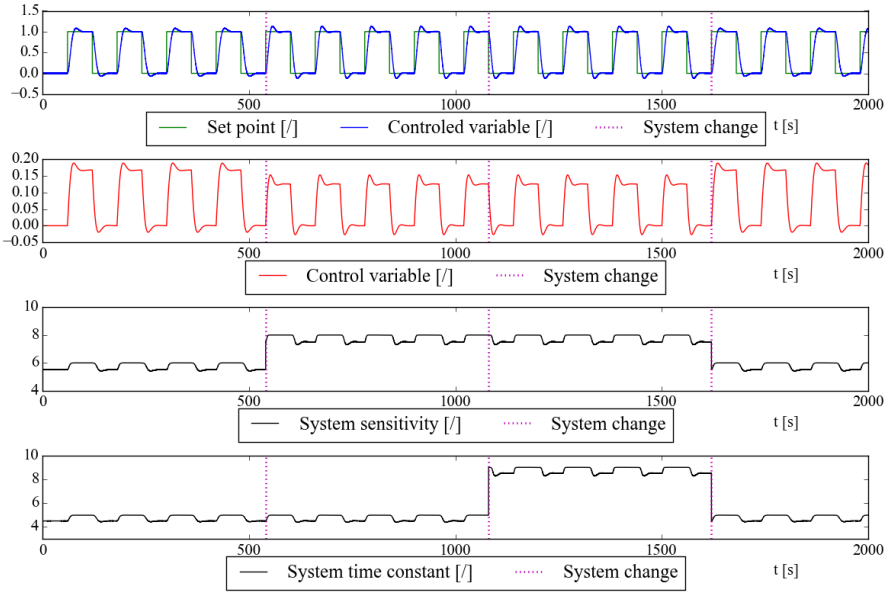

Figure 5.18: Data used for the experiment - output of the simulated system. The plot is adopted from study [mc3].

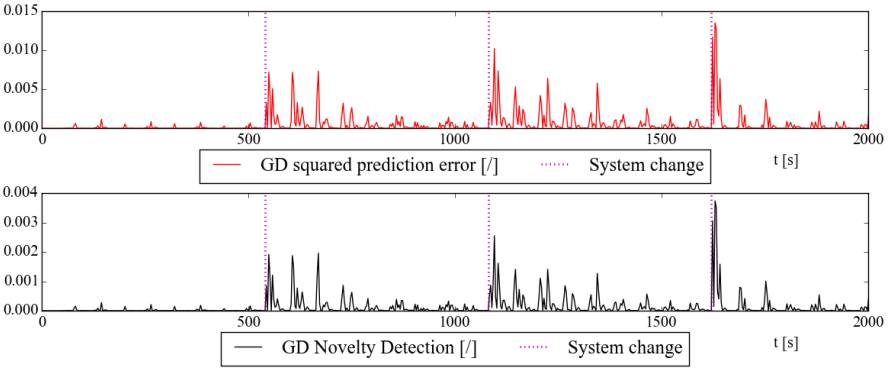

Figure 5.19: Novelty detection results with the LMS algorithm. The plot is adopted from study [mc3].

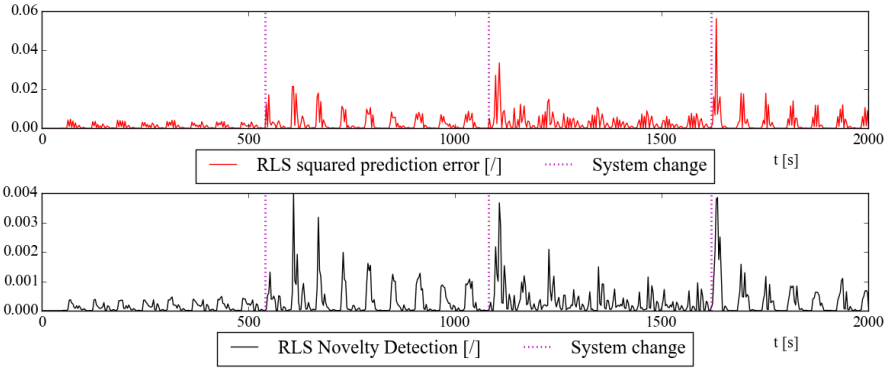

Figure 5.20: Novelty detection results with the RLS algorithm. The plot is adopted from study [mc3].

were changed randomly every 200 samples. These changes are sharp and with unit standard deviation and zero mean. The data contains 2000 of such changes (total length of data is 400000 samples).

The signal to noise ratio (SNR) was evaluated for every experiment to measure how the classifier performance declines with the increasing level of noise in the data. Because the data and also the noise have zero mean, than it is possible to estimate the SNR with formula as follows

$$\mathrm{SNR} = 10 \log_{10} \frac{\sigma_y^2}{\sigma_v^2} \tag{5.9}$$

where σ_y is the standard deviation of unknown system output and σ_v is the standard deviation of noise $v(k)$. The classification performance was evaluated for data contaminated with three different types of noise.

White Gaussian noise This is a noise with normal distribution. The Gaussian noise used in this study has zero mean value and the standard deviation was altered to simulate different levels of SNR.

White uniform noise This noise has an uniform distribution. It is used in this study with different ranges of values to achieve various levels of SNR. Although this type of noise seems to be unnatural, a noise similar to this one can be produced by a process of uniform quantisation in real applications [92].

Brownian noise The alternative name of this noise is the random-walk noise. This noise was obtained by integration of white Gaussian noise. To prevent the noise signal from wandering off during long integration, the leaky integration was used. The leak of 1% was enough to keep the noise in range where the adaptive model still works effectively. Such a small leak cuts only the lowest frequencies, so it does not influence the results of experiment.

5.3.2.2 Results

The results were obtained as an average of 5 simulations with different random seed. The AUROC and maximum accuracy were used as the criteria of classifier performance. Although both measures are likely to be correlated, they provide different information. The AUROC is equal to the probability that a classifier will rank a

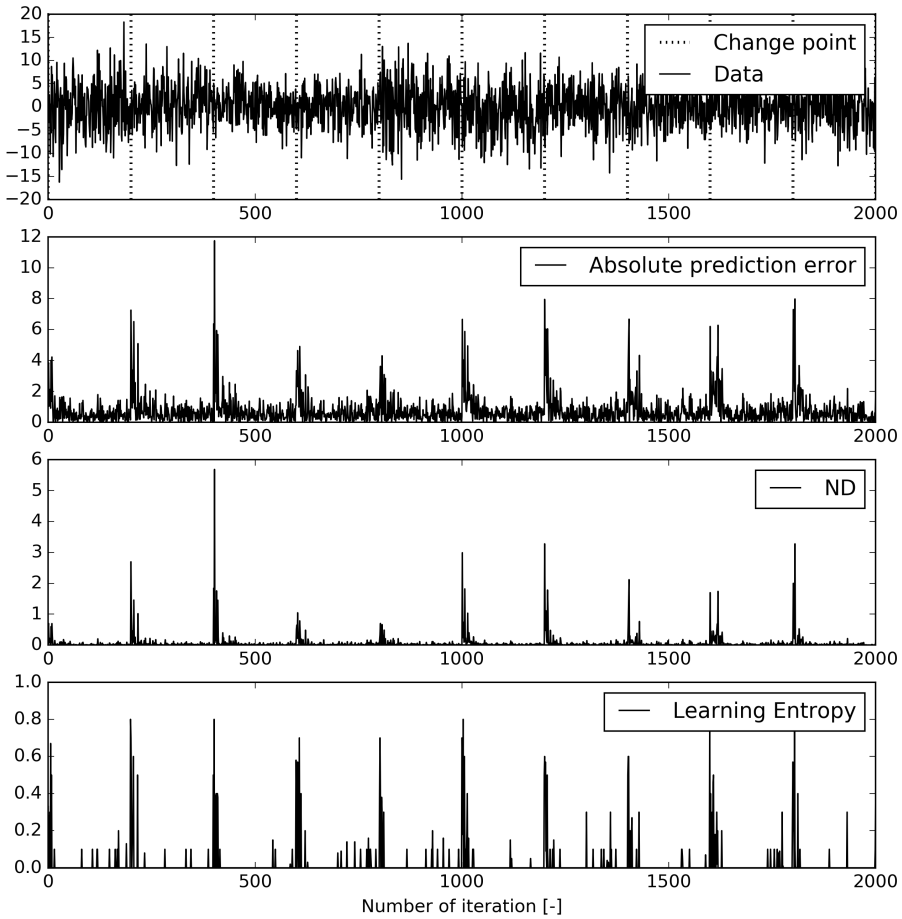

Figure 5.21: Demonstration how the used algorithms process the data (annotate novelty). This Figure is adopted from [mc4].

randomly chosen positive instance higher than a randomly chosen negative example. On the other hand, the maximum accuracy reflects the best result what is possible to achieve if the criteria is selected correctly. It is important to note, that both used criteria (AUROC and maximal accuracy) are build on the assumption that there is the same cost for the false positives and for the false negatives.

A demonstration how the used algorithms annotate the novelty in data is shown in Figure 5.21. Optimal result can be described as follows:

- high values of novelty on change point positions,
- low values of novelty elsewhere,
- all high values (detection of change) should have similar height (for easy selection of threshold).

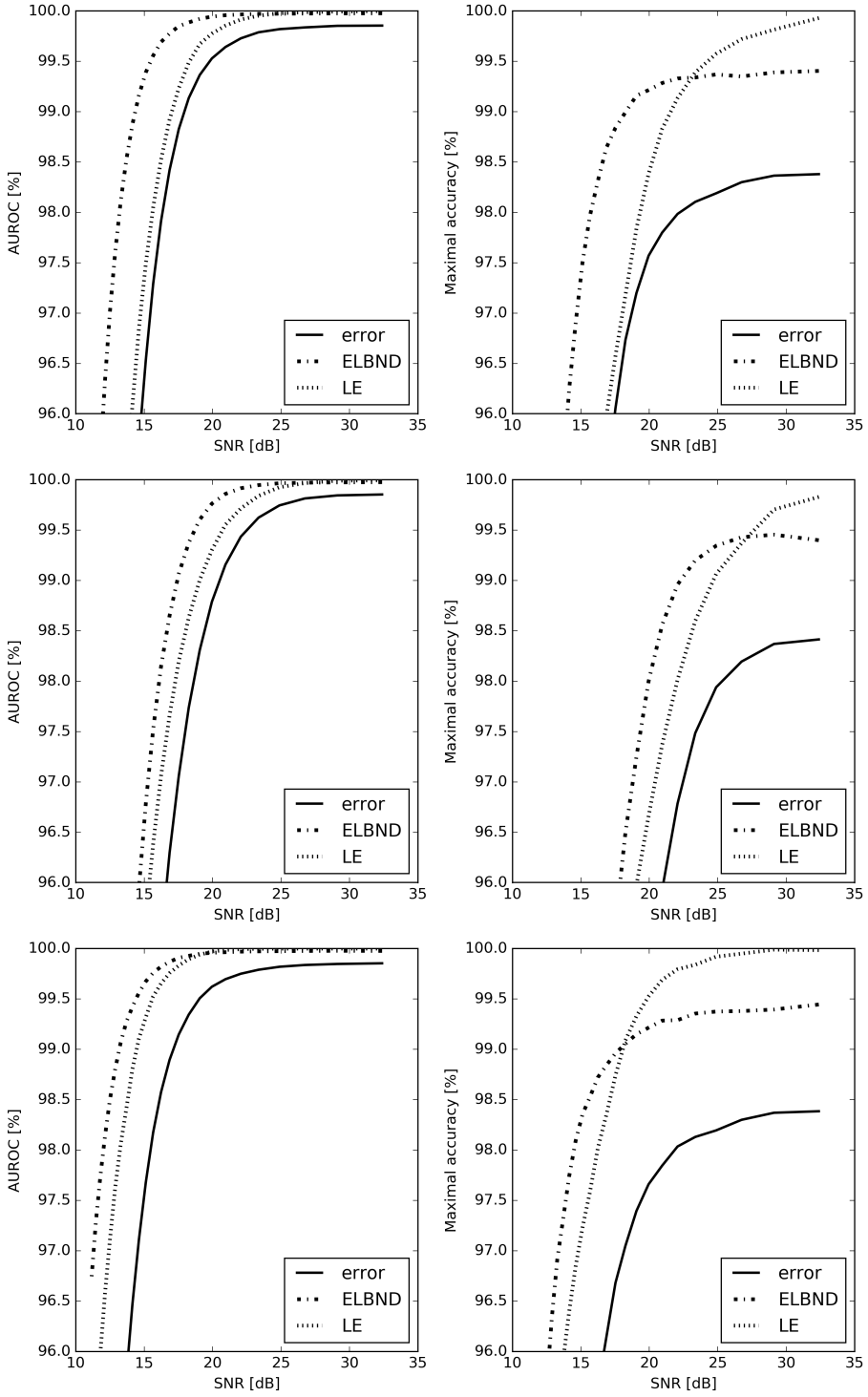

Figure 5.22: AUROC and maximal accuracy of classifiers using RLS adaptive algorithm with different noise distribution, from top: normal, Brownian, uniform. The *error* label stands for accuracy based only on error. This Figure is adopted from [mc4].

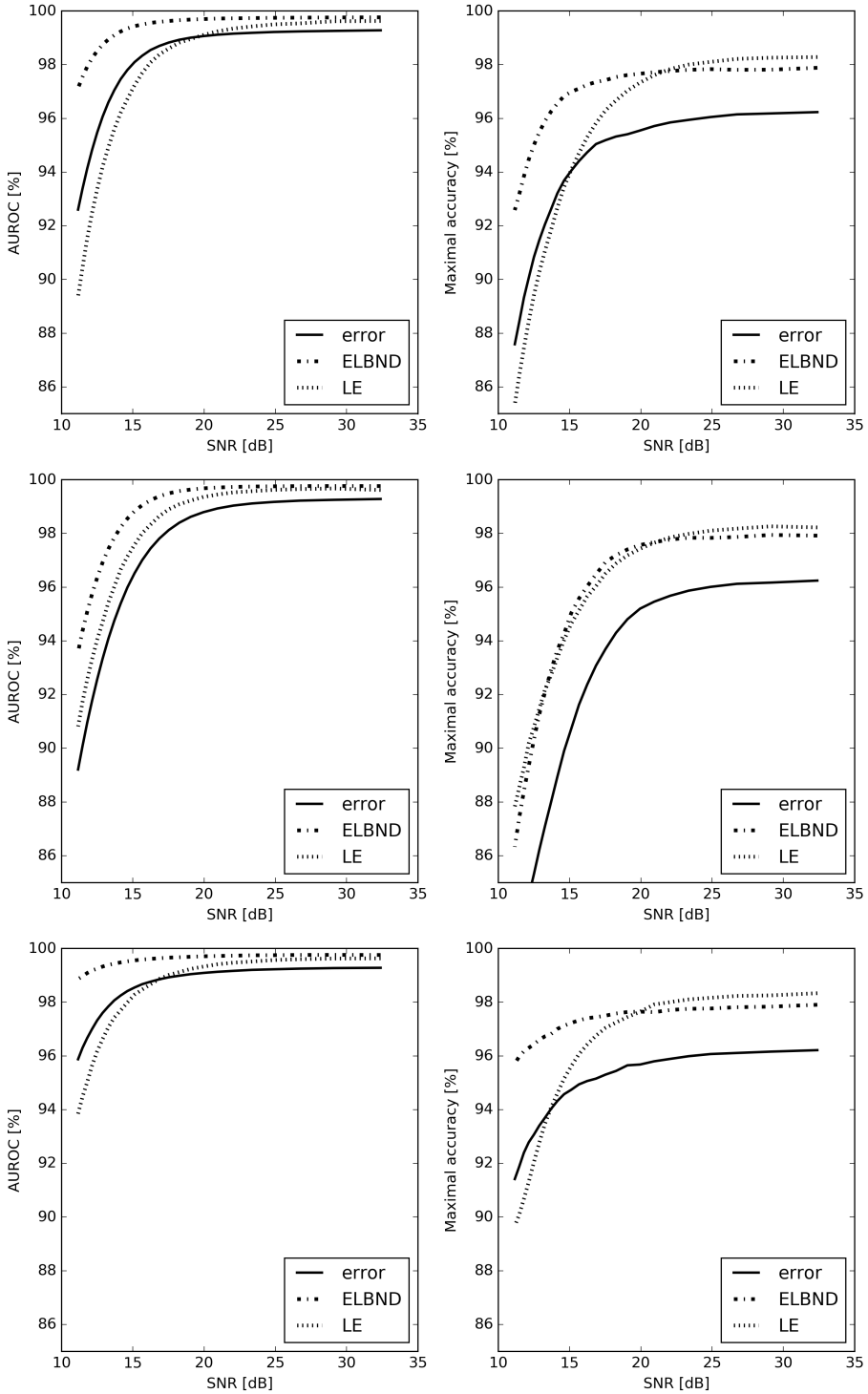

Figure 5.23: AUROC and maximal accuracy of classifiers using NLMS adaptive algorithm with different noise distribution, from top: normal, Brownian, uniform. The *error* label stands for accuracy based only on error. This Figure is adopted from [mc4].

The obtained results are shown in Figure 5.23 for the NLMS algorithm, and in Figure 5.22 for the RLS algorithm. The three lines in figures represents the results of ELBND, LE and reference (detection based only on error of adaptive model).

5.3.2.3 Conclusion

The study presented in this subsection compares two adaptive novelty detection methods (LE, ELBND) implemented on two different adaptive filters (NLMS, RLS). The metric used for comparison was AUROC and maximum accuracy of tested methods during classification of simulated process changes. This study results can be concluded as:

- LE and ELBND were always beneficial in comparison to classification based only on the error of the adaptive model.
- For high level of noise (low SNR) the ELBND scored always better than LE.
- For low level of noise (high SNR) the LE scored always better than ELBND in maximal accuracy.
- Performance of all detectors were generally better when RLS adaptation was applied.

5.3.3 ELBND time complexity analysis

This subsection presents results from study [mc5]. The study investigate time complexity of algorithms for adaptive novelty detection. One of the studied algorithms was ELBND. The other studied algorithms are: LE, Mahanobilis distance of weights increments (MD) [45] and Fuzzy Density (FD) of weights increments [mc5].

The time complexity of the ELBND algorithm iteration is broken down step by step in Tab 5.7. As you can see from the table, the complexity of the algorithm strongly relies on the target device, environment and language of the chosen implementation.

The final measured results from comparison of the ELBND and the other similar methods is possible to see in Tab. 5.8.

order	operation	complexity	additions	multiplications	note
1.	$o_1 = \mathbf{\Delta w}(\mathbf{k})e$	$O(n)$	0	n	-
2.	$o_2 = \|o_1\|$	$O(n)$	0	0	abs()
3.	$\max(o_2)$	$O(n)$	0	0	max()

Table 5.7: Time complexity and number of operations for one iteration of ELBND algorithms, n is the number of adaptive model parameters. The table is from [mc5]

n	ELBND	LE	FD	MD
3	0.040565	12.106009	17.114869	27.900899
13	0.057102	12.484739	91.514165	45.028556
23	0.067401	12.566594	132.167673	51.513525
33	0.078859	12.785707	170.275357	60.336528
43	0.091092	13.055424	211.303988	71.177681
53	0.102716	13.517995	251.945065	82.149545
63	0.112905	13.724406	291.961026	95.538772
73	0.128387	14.128523	330.522018	108.705011
83	0.140568	14.402364	370.774985	128.212707
93	0.152008	14.572935	409.528680	146.476234

Table 5.8: Measured time for all algorithms in miliseconds.

As the results shows, the ELBND is much faster than the other methods. Especially if the big number of adaptive weights is required for evaluation.

Chapter 6

Conclusion

In this thesis the derivation, implementation and experimental analysis of newly developed adaptive novelty detection method - ELBND is described. The method is designed to by used with any supervised adaptive algorithm that has adaptive parameters and error. The experimental analysis that is present in this work features adaptive filters as the adaptive models used together with ELBND. Although the adaptive filters are one of the simplest adaptive algorithms, the ELBND algorithm is able to effectively utilize information produced by their operation.

As shown in previous chapters, the ELBND method with adaptive filters does not require the knowledge of future samples (whole batches) [mc2, mc5, mc1, mc10]. Thus the ELBND is suitable for online data streams processing. Therefore *the 1. goal of this thesis is accomplished*. This is the key feature of the ELBND method, because not many novelty detection methods are designed to be able work in this way. Even though this feature is required by a lot of common novelty detection applications.

Yet another important aspect of the ELBND method is the requirement for low computational power [mc2, mc5, mc6, mc9]. The time complexity of the ELBND method is only constant. In other words, the constant time complexity is the lowest time complexity possible to have for an algorithm. Also it is shown that the particular count of machine instructions necessary to estimate the level of novelty with ELBND is small. Furthermore the experimental analysis prove that the ELBND algorithm is

fast in comparison with alternative state of the art algorithms. *As a conclusion the ELBND overall speed is good enough to accomplish the 2. goal of this thesis.* The speed aspect of the ELBND together with its ability to work without data batches makes it perfect option for real time novelty detection applications.

The ELBND is an adaptive method and this fact should ensure some concept drift robustness by itself. Because the ELBND method use adaptive parameter increments and model error as the input values, it is possible to improve its robustness furthermore with the selection of an adaptive model. If the adaptive model is designed to handle data offsets and other non-stationarity in a smart way, then the penalization for ill conditioned data can be reduced. The ELBND ability to perform well with distorted data was investigated more via means of experimental analysis [mc4, mc7, mc9]. Although it is difficult to measure this robustness in some fair way, the results indicate that the ELBND posses ability to deal with reasonably sized concept drift. Even when tested with adaptive models derived with assumption of zero-mean data. *Therefore the 3. goal of this thesis is also accomplished.* According to the results presented, the ELBND can be considered as an algorithm safe for operation on data with reasonable sized gradual or recurring concept drift.

Furthermore, the presented method - ELBND - is different from all other published methods and that is the main reason why the development of this method is a contribution to the field of machine learning and signal processing.

Chapter 7

References

[1] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, p. 15, 2009.

[2] V. Hodge and J. Austin, "A survey of outlier detection methodologies," *Artificial intelligence review*, vol. 22, no. 2, pp. 85–126, 2004.

[3] J. Gertler, *Fault Detection and Diagnosis*. Springer, 2015.

[4] M. Masud, J. Gao, L. Khan, J. Han, and B. M. Thuraisingham, "Classification and novel class detection in concept-drifting data streams under time constraints," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 6, pp. 859–874, 2011.

[5] A. Dries and U. Rückert, "Adaptive concept drift detection," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 2, no. 5-6, pp. 311–327, 2009.

[6] L. Aguayo and G. A. Barreto, "Novelty detection in time series using self-organizing neural networks: A comprehensive evaluation," *Neural Processing Letters*, pp. 1–28, 2017.

[7] S. Ahmad, A. Lavin, S. Purdy, and Z. Agha, "Unsupervised real-time anomaly detection for streaming data," *Neurocomputing*, vol. 262, pp. 134 – 147, 2017.

[8] M. Markou and S. Singh, "Novelty detection: a review—part 1: statistical approaches," *Signal processing*, vol. 83, no. 12, pp. 2481–2497, 2003.

[9] D. H. Wolpert, W. G. Macready *et al.*, "No free lunch theorems for optimization," *IEEE transactions on evolutionary computation*, vol. 1, no. 1, pp. 67–82, 1997.

[10] M. A. Pimentel, D. A. Clifton, L. Clifton, and L. Tarassenko, "A review of novelty detection," *Signal Processing*, vol. 99, pp. 215–249, 2014.

[11] R. McGill, J. W. Tukey, and W. A. Larsen, "Variations of box plots," *The American Statistician*, vol. 32, no. 1, pp. 12–16, 1978.

[12] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, 1943.

[13] M. Markou and S. Singh, "Novelty detection: a review—part 2: neural network based approaches," *Signal processing*, vol. 83, no. 12, pp. 2499–2521, 2003.

[14] S. Haykin, *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1994.

[15] J. P. Egan, "Signal detection theory and roc analysis," 1975.

[16] F. E. Grubbs, "Procedures for detecting outlying observations in samples," *Technometrics*, vol. 11, no. 1, pp. 1–21, 1969.

[17] C. C. Aggarwal and P. S. Yu, "Outlier detection with uncertain data," in *Proceedings of the 2008 SIAM International Conference on Data Mining*. SIAM, 2008, pp. 483–493.

[18] R. Ruotolo and C. Surace, "A statistical approach to damage detection through vibration monitoring," *Applied mechanics in the Americas*, pp. 314–317, 1997.

[19] H. E. Solberg and A. Lahti, "Detection of outliers in reference distributions: performance of horn's algorithm," *Clinical chemistry*, vol. 51, no. 12, pp. 2326–2332, 2005.

[20] V. Barnett and T. Lewis, "Outliers in statistical data: Wiley series in probability and statistics," 1994.

[21] L. K. Hansen, S. Sigurdsson, T. Kolenda, F. A. Nielsen, U. Kjems, and J. Larsen, "Modeling text with generalizable gaussian mixtures," in *icassp*. IEEE, 2000, pp. 3494–3497.

[22] L. Tarassenko, P. Hayton, N. Cerneaz, and M. Brady, "Novelty detection for the identification of masses in mammograms," 1995.

[23] R. O. Duda, P. E. Hart, D. G. Stork *et al.*, *Pattern classification*. Wiley New York, 1973, vol. 2.

[24] D.-Y. Yeung and Y. Ding, "Host-based intrusion detection using dynamic and static behavioral models," *Pattern recognition*, vol. 36, no. 1, pp. 229–243, 2003.

[25] S.-B. Cho and H.-J. Park, "Efficient anomaly detection by modeling privilege flows using hidden markov model," *Computers & Security*, vol. 22, no. 1, pp. 45–55, 2003.

[26] Y. Xie and S.-Z. Yu, "A large-scale hidden semi-markov model for anomaly detection on user browsing behaviors," *Networking, IEEE/ACM Transactions on*, vol. 17, no. 1, pp. 54–65, 2009.

[27] J. Hu, X. Yu, D. Qiu, and H.-H. Chen, "A simple and efficient hidden markov model scheme for host-based anomaly intrusion detection," *Network, IEEE*, vol. 23, no. 1, pp. 42–47, 2009.

[28] S. S. Joshi and V. V. Phoha, "Investigating hidden markov models capabilities in anomaly detection," in *Proceedings of the 43rd annual Southeast regional conference-Volume 1*. ACM, 2005, pp. 98–103.

[29] T. V. Duong, H. H. Bui, D. Q. Phung, and S. Venkatesh, "Activity recognition and abnormality detection with the switching hidden semi-markov model," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 838–845.

[30] E. L. Andrade, S. Blunsden, and R. B. Fisher, "Hidden markov models for optical flow analysis in crowds," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 1. IEEE, 2006, pp. 460–463.

[31] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[32] B. Schölkopf and A. J. Smola, *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

[33] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in *Advances in neural information processing systems*, 2000, pp. 582–588.

[34] A. B. Gardner, A. M. Krieger, G. Vachtsevanos, and B. Litt, "One-class novelty detection for seizure analysis from intracranial eeg," *The Journal of Machine Learning Research*, vol. 7, pp. 1025–1044, 2006.

[35] G. Li, C. Wen, and Z. Li, "A new online learning with kernels method in novelty detection," in *IECON 2011-37th Annual Conference on IEEE Industrial Electronics Society*. IEEE, 2011, pp. 2311–2316.

[36] C. P. Diehl and J. B. Hampshire, "Real-time object classification and novelty detection for collaborative video surveillance," in *Neural Networks, 2002. IJCNN'02. Proceedings of the 2002 International Joint Conference on*, vol. 3. IEEE, 2002, pp. 2620–2625.

[37] D. M. Tax and R. P. Duin, "Support vector data description," *Machine learning*, vol. 54, no. 1, pp. 45–66, 2004.

[38] M. Wu and J. Ye, "A small sphere and large margin approach for novelty detection using training data with outliers," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 11, pp. 2088–2092, 2009.

[39] T. Le, D. Tran, W. Ma, and D. Sharma, "An optimal sphere and two large margins approach for novelty detection," in *Neural Networks (IJCNN), The 2010 International Joint Conference on*. IEEE, 2010, pp. 1–6.

[40] Y. Xiao, B. Liu, L. Cao, X. Wu, C. Zhang, Z. Hao, F. Yang, and J. Cao, "Multi-sphere support vector data description for outliers detection on multi-distribution data," in *Data Mining Workshops, 2009. ICDMW'09. IEEE International Conference on*. IEEE, 2009, pp. 82–87.

[41] T. Le, D. Tran, W. Ma, and D. Sharma, "Multiple distribution data description learning algorithm for novelty detection," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2011, pp. 246–257.

[42] X. Peng and D. Xu, "Efficient support vector data descriptions for novelty detection," *Neural Computing and Applications*, vol. 21, no. 8, pp. 2023–2032, 2012.

[43] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[44] V. Hautamaki, I. Karkkainen, and P. Franti, "Outlier detection using k-nearest neighbour graph," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 3. IEEE, 2004, pp. 430–433.

[45] P. C. Mahalanobis, "On the generalized distance in statistics." National Institute of Science of India, 1936.

[46] F. Angiulli and C. Pizzuti, "Fast outlier detection in high dimensional spaces," in *European Conference on Principles of Data Mining and Knowledge Discovery*. Springer, 2002, pp. 15–27.

[47] Y. Liao and V. R. Vemuri, "Use of k-nearest neighbor classifier for intrusion detection," *Computers & Security*, vol. 21, no. 5, pp. 439–448, 2002.

[48] Q. P. He and J. Wang, "Fault detection using the k-nearest neighbor rule for semiconductor manufacturing processes," *Semiconductor manufacturing, IEEE transactions on*, vol. 20, no. 4, pp. 345–354, 2007.

[49] K. P. F.R.S., "Liii. on lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.

[50] T. Kohonen, "Self-organizing maps, vol. 30 of springer series in information sciences," 2001.

[51] T. Harris, "Neural network in machine health monitoring," *Professional Engineering*, 1993.

[52] V. Emamian, M. Kaveh, and A. H. Tewfik, "Robust clustering of acoustic emission signals using the kohonen network," in *Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on*, vol. 6. IEEE, 2000, pp. 3891–3894.

[53] S. Singh and M. Markou, "An approach to novelty detection applied to the classification of image regions," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 16, no. 4, pp. 396–407, 2004.

[54] C. Oswald, "Provozní ekonomicko-ekologická optimalizace při regulaci biomasových kotlů," Ph.D. dissertation, České vysoké učení technické v Praze, Fakulta strojní, Ústav přísstrojové a řídicí techniky, 2017.

[55] C.-Y. Liou, W.-C. Cheng, J.-W. Liou, and D.-R. Liou, "Autoencoder for words," *Neurocomputing*, vol. 139, pp. 84–96, 2014.

[56] H. A. Dau, V. Ciesielski, and A. Song, "Anomaly detection using replicator neural networks trained on examples of one class," in *Asia-Pacific Conference on Simulated Evolution and Learning*. Springer, 2014, pp. 311–322.

[57] J. Elman, "Generalization, simple recurrent networks, and the emergence of structure," in *Proceedings of the twentieth annual conference of the Cognitive Science Society*. Mahwah, NJ: Lawrence Erlbaum Associates, 1998, p. 6.

[58] M. Augusteijn and B. Folkert, "Neural network classification and novelty detection," *International Journal of Remote Sensing*, vol. 23, no. 14, pp. 2891–2902, 2002.

[59] M. D. Richard and R. P. Lippmann, "Neural network classifiers estimate bayesian a posteriori probabilities," *Neural computation*, vol. 3, no. 4, pp. 461–483, 1991.

[60] I. Bukovsky, "Learning Entropy: Multiscale Measure for Incremental Learning," *Entropy*, vol. 15, no. 10, pp. 4159–4187, Sep. 2013, 00004. [Online]. Available: http://www.mdpi.com/1099-4300/15/10/4159/

[61] C. M. Bishop, "Novelty detection and neural network validation," in *Vision, Image and Signal Processing, IEE Proceedings-*, vol. 141. IET, 1994, pp. 217–222.

[62] I. Diaz and J. Hollmen, "Residual generation and visualization for understanding novel process conditions," in *Neural Networks, 2002. IJCNN'02. Proceedings of the 2002 International Joint Conference on*, vol. 3. IEEE, 2002, pp. 2070–2075.

[63] J. J. Gertler, "Survey of model-based failure detection and isolation in complex plants," *Control Systems Magazine, IEEE*, vol. 8, no. 6, pp. 3–11, 1988.

[64] C. G. Healey, K. S. Booth, and J. T. Enns, "Visualizing real-time multivariate data using preattentive processing," *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, vol. 5, no. 3, pp. 190–221, 1995.

[65] G. Williams, R. Baxter, H. He, S. Hawkins, and L. Gu, "A comparative study of rnn for outlier detection in data mining," in *null*. IEEE, 2002, p. 709.

[66] S. Hawkins, H. He, G. Williams, and R. Baxter, "Outlier detection using replicator neural networks," in *Data warehousing and knowledge discovery*. Springer, 2002, pp. 170–180.

[67] G. E. Hinton and J. A. Anderson, *Parallel Models of Associative Memory: Updated Edition*. Psychology Press, 2014.

[68] A. H. Sayed, *Fundamentals of adaptive filtering*. John Wiley & Sons, 2003.

[69] D. P. Mandic, "A generalized normalized gradient descent algorithm," *Signal Processing Letters, IEEE*, vol. 11, no. 2, pp. 115–118, 2004.

[70] V. S. A. Kumar *et al.*, "Comparison of stable nlmf and nlms algorithms for adaptive noise cancellation in ecg signal with gaussian, binary and uniform sig-

nals as inputs," *International Journal of Engineering Research and Applications*, vol. 4, no. 8, pp. 28–33, 2014.

[71] V. H. Nascimento and J. C. M. Bermudez, "Probability of divergence for the least-mean fourth algorithm," *IEEE transactions on signal processing*, vol. 54, no. 4, pp. 1376–1385, 2006.

[72] P. I. Hubscher and J. C. M. Bermudez, "An improved statistical analysis of the least mean fourth (lmf) adaptive algorithm," *IEEE transactions on Signal Processing*, vol. 51, no. 3, pp. 664–671, 2003.

[73] E. Niedermeyer and F. L. da Silva, *Electroencephalography: basic principles, clinical applications, and related fields*. Lippincott Williams & Wilkins, 2005.

[74] S. Baillet, J. C. Mosher, and R. M. Leahy, "Electromagnetic brain mapping," *IEEE Signal processing magazine*, vol. 18, no. 6, pp. 14–30, 2001.

[75] M. J. Goldman and N. Goldschlager, *Principles of clinical electrocardiography*. Lange Medical Los Altos, CA, 1970, vol. 1973.

[76] R. Elwell and R. Polikar, "Incremental learning of concept drift in nonstationary environments," *IEEE Transactions on Neural Networks*, vol. 22, no. 10, pp. 1517–1531, 2011.

[77] C. Alippi, G. Boracchi, and M. Roveri, "An effective just-in-time adaptive classifier for gradual concept drifts," in *Neural Networks (IJCNN), The 2011 International Joint Conference on*. IEEE, 2011, pp. 1675–1682.

[78] J. C. Schlimmer and R. H. Granger, "Beyond incremental processing: Tracking concept drift." in *AAAI*, 1986, pp. 502–507.

[79] T. Lane and C. E. Brodley, "Approaches to online learning and concept drift for user identification in computer security." in *KDD*, 1998, pp. 259–263.

[80] L. L. Minku, A. P. White, and X. Yao, "The impact of diversity on online ensemble learning in the presence of concept drift," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 5, pp. 730–742, 2010.

[81] A. Tsymbal, "The problem of concept drift: definitions and related work," *Computer Science Department, Trinity College Dublin*, vol. 106, no. 2, 2004.

[82] A. M. Narasimhamurthy and L. I. Kuncheva, "A framework for generating data to simulate changing environments." in *Artificial Intelligence and Applications*, 2007, pp. 415–420.

[83] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Computing Surveys (CSUR)*, vol. 46, no. 4, p. 44, 2014.

[84] S. J. Delany, P. Cunningham, A. Tsymbal, and L. Coyle, "A case-based technique for tracking concept drift in spam filtering," *Knowledge-Based Systems*, vol. 18, no. 4, pp. 187–195, 2005.

[85] I. Bukovsky and C. Oswald, "Case study of learning entropy for adaptive novelty detection in solid-fuel combustion control," in *Intelligent Systems in Cybernetics and Automation Theory*. Springer, 2015, pp. 247–257.

[86] A. M. Narasimhamurthy and L. I. Kuncheva, "A framework for generating data to simulate changing environments." in *Artificial Intelligence and Applications*, 2007, pp. 415–420.

[87] I. Bukovsky, "Learning entropy: Multiscale measure for incremental learning," *Entropy*, vol. 15, no. 10, pp. 4159–4187, 2013.

[88] J. S. Richman and J. R. Moorman, "Physiological time-series analysis using approximate entropy and sample entropy," *American Journal of Physiology. Heart and Circulatory Physiology*, vol. 278, no. 6, pp. H2039–2049, Jun. 2000.

[89] D. Bai, T. Qiu, and X. Li, "The sample entropy and its application in eeg based epilepsy detection," *Sheng wu yi xue gong cheng xue za zhi= Journal of biomedical engineering= Shengwu yixue gongchengxue zazhi*, vol. 24, no. 1, pp. 200–205, 2007.

[90] D. E. Lake, J. S. Richman, M. P. Griffin, and J. R. Moorman, "Sample entropy analysis of neonatal heart rate variability," *American Journal of Physiology-Regulatory, Integrative and Comparative Physiology*, vol. 283, no. 3, pp. R789–R797, 2002.

[91] P. McSharry and G. Clifford, "Ecgsyn-a realistic ecg waveform generator," *URL http://www. physionet. org/physiotools/ecgsyn*, 2003.

[92] B. Widrow, I. Kollar, and M.-C. Liu, "Statistical theory of quantization," *IEEE Transactions on instrumentation and measurement*, vol. 45, no. 2, pp. 353–361, 1996.

Author's references

[mc1] M. Cejnek, P. M. Benes, and I. Bukovsky, "Another adaptive approach to novelty detection in time series." Fourth International conference on Computer Science and Information Technology, Academy and Industry Research Collaboration Center (AIRCC), 2014, pp. 341–351, doi: 10.5121/csit.2014.4229, ISBN 78-1-921987-27-4.

[mc2] M. Cejnek, "Rychlé algoritmy pro adaptivní detekci novosti," in *Nové metody a postupy v oblasti přístrojové techniky, automatického řízení a informatiky 2018*. České vysoké učení technické v Praze, 2018, pp. 80 – 90, ISBN 978-80-01-06477-1.

[mc3] C. Oswald, M. Cejnek, J. Vrba, and I. Bukovsky, "Novelty detection in system monitoring and control with honu," in *Applied Artificial Higher Order Neural Networks for Control and Recognition*, M. Zhang, Ed. IGI Global, 2016, pp. 61–78.

[mc4] M. Cejnek and I. Bukovsky, "Influence of type and level of noise on the performance of an adaptive novelty detector," in *2017 IEEE 16th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC)*. IEEE, 2017, pp. 373–377, ISBN 978-1-5386-0771-8.

[mc5] M. Cejnek and A. Peichl, "Rychlost adaptivní algoritmů pro detekce novosti," in *Nové metody a postupy v oblasti přístrojové techniky, automatického řízení a informatiky 2018*. České vysoké učení technické v Praze, 2018, pp. 91 – 99, ISBN 978-80-01-06477-1.

[mc6] M. Cejnek, I. Bukovsky, N. Homma, and O. Liska, "Adaptive polynomial filters with individual learning rates for computationally efficient lung tumor motion prediction," in *Computational Intelligence for Multimedia Understanding, 2015 International Workshop on.* IEEE, 2015, pp. 1–5, ISBN 9781467384582.

[mc7] M. Cejnek and I. Bukovsky, "Online data centering modifications for adaptive filtering with nlms algorithm," in *Neural Networks (IJCNN), 2016 International Joint Conference on.* IEEE, 2016, pp. 1767–1771, ISBN 978-1-5090-0620-5.

[mc8] M. Cejnek, I. Bukovsky, and O. Vysata, "Adaptive classification of eeg for dementia diagnosis," in *Computational Intelligence for Multimedia Understanding (IWCIM), 2015 International Workshop on.* IEEE, 2015, pp. 1–5, ISBN 9781467384582.

[mc9] M. Cejnek and I. Bukovsky, "Concept drift robust adaptive novelty detection for data streams," *Neurocomputing*, vol. 309, pp. 46 – 53, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231218305253

[mc10] I. Bukovsky, M. Cejnek, J. Vrba, and N. Homma, "Study of learning entropy for onset detection of epileptic seizures in eeg time series," in *Neural Networks (IJCNN), 2016 International Joint Conference on.* IEEE, 2016, pp. 3302–3305, ISBN 978-1-5090-0620-5.