



**VYSOKÁ ŠKOLA
CHEMICKO-TECHNOLOGICKÁ
V PRAZE**

Fakulta chemicko-inženýrská

Ústav počítačové a řídicí techniky

POUŽITÍ ADAPTIVNÍCH SYSTÉMŮ PŘI ANALÝZE DAT

DISERTAČNÍ PRÁCE

AUTOR

Ing. Jan Vrba

ŠKOLITEL

doc. Ing. Jan Mareš, Ph.D.

ŠKOLITEL SPECIALISTA

doc. Ing. Pavel Hrnčířík, Ph.D.

STUDIJNÍ PROGRAM

Chemické a procesní inženýrství (čtyřleté)

STUDIJNÍ OBOR

Technická kybernetika

ROK

2020



**UNIVERSITY OF
CHEMISTRY AND TECHNOLOGY
PRAGUE**

Faculty of Chemical Engineering

Department of Computing and Control Engineering

ADAPTIVE SYSTEMS IN DATA ANALYSIS

DISSERTATION

AUTHOR

Ing. Jan Vrba

SUPERVISOR

doc. Ing. Jan Mareš, Ph.D.

SUPERVISOR SPECIALIST

doc. Ing. Pavel Hrnčířík, Ph.D.

STUDY PROGRAMME

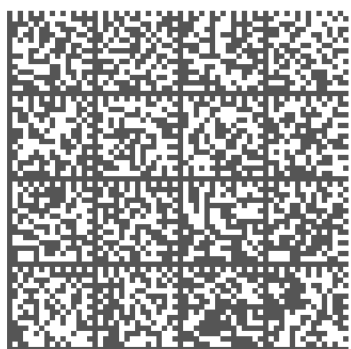
Chemical and Process Engineering

FIELD OF STUDY

Technical Cybernetics

YEAR

2020



Tato disertační práce byla vypracována na Ústavu počítačové a řídicí techniky v období září 2016 – říjen 2020.

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že na moji práci se vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon). Zejména se jedná o skutečnost, že Vysoká škola chemicko-technologická v Praze, popř. jiné vzdělávací zařízení, ve kterém jsem svou práci vypracoval, má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona. Pokud bych v budoucnu poskytl licenci o užití práce jinému subjektu, je Vysoká škola chemicko-technologická v Praze, popř. jiné vzdělávací zařízení, ve kterém jsem svou práci vypracoval, oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložil a to podle okolností až do jejich skutečné výše.

Souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách, ve znění pozdějších předpisů.

V Praze dne

.....

Ing. Jan Vrba

Poděkování

Děkuji vedoucímu mé dizertační práce doc. Ing. Janu Marešovi, Ph.D. za všestrannou pomoc, podporu a trpělivost. Děkuji doc. Ing. Ivo Bukovskému, Ph.D. za otevření dveří do světa detekce novosti. Rád bych poděkoval také Matouši Cejnkovi za inspiraci, blahodárné diskuze a spolupráci nejen na bitevním poli světa H&G. Děkuji všem zaměstnancům Ústavu počítačové a řídicí techniky na VŠCHT. V neposlední řadě bych chtěl poděkovat také všem přátelům, kteří mi pomohli udržet zdravou životní rovnováhu. Zvláštní poděkování patří mým rodičům a sestře Radaně za podporu během mého dlouhého studia. Děkuji také Otovi, který mi přinesl tolik štěstí a radosti. Nejvíce děkuji mé Kazumi za podporu, trpělivost, obětavost a za vytvoření ideálních podmínek, ve kterých jsem mohl práci psát. Děkuji.

Souhrn

Dizertační práce se zabývá použitím adaptivních systémů v oblasti detekce novosti. Tento přístup v oblasti detekce novosti v datech se stal v posledních letech slibným směrem výzkumu. V rámci této práce je navržen nový algoritmus pojmenovaný jako Extreme Seeking Entropy. Tento algoritmus je založen na vyhodnocování přírůstku adaptivních parametrů systémů pomocí zobecněného Paretova rozdělení. Navržený algoritmus byl otestován na celé řadě typů syntetických dat, která reprezentují různé druhy novosti. Pro detekci změny trendu a skokové změny generátoru signálu pak bylo provedeno i vyhodnocení úspěšnosti detekce. Dále byla provedena experimentální studie zabývající se časovou náročností výpočtu algoritmu, vyhodnocena ROC (Receiver Operating Characteristic) křivka pro detekci změny trendu, a provedena studie detekce epilepsie v záznamu EEG myši.

Klíčová slova

adaptivní systémy, detekce novosti, časové řady, extreme seeking entropy

Summary

The dissertation deals with the use of adaptive systems in the field of novelty detection. The use of adaptive systems for detecting novelty in data has become a promising direction of research in recent years. In this work, a new algorithm was designed using adaptive systems, called Extreme Seeking Entropy. This algorithm is based on evaluating the increment of adaptive parameters of systems using a generalized Pareto distribution. The proposed algorithm has been tested on a number of types of synthetic data that represent different types of novelty. To detect a change in the trend and a step-change in the signal generator, an evaluation of the detection success was performed. Furthermore, an experimental study was performed dealing with the time consumption of the algorithm, the ROC curve was evaluated to detect a change in the trend, and a study was performed to detect epilepsy in the EEG mouse record.

Keywords

adaptive systems, novelty detection, time series, extreme seeking entropy

Obsah

1	Úvod	1
1.1	Cíle dizertační práce	1
1.2	Členění práce	2
2	Číslicové adaptivní filtry a algoritmy	3
2.1	Adaptivní filtry	5
2.1.1	Lineární FIR filtry	6
2.1.2	Volterrovy filtry	7
2.1.3	Polynomiální neuronové jednotky	8
2.1.4	Fuzzy filtry	9
2.2	Adaptivní algoritmy	11
2.2.1	Algoritmy LMS a NLMS	11
2.2.2	Algoritmus RLS	13
2.2.3	Algoritmus Generalized Normalized Gradient Descent	15
2.2.4	Gradient descent pro fuzzy filtry	15
3	Přehled metod detekce novosti	19
3.1	Vybrané statistické metody	20
3.1.1	Metody využívající Gaussovy smíšené modely	20
3.1.2	Metody založené na teorii extrémních hodnot	21
3.1.3	Markovovy skryté modely	22
3.2	Vybrané metody využívající strojové učení a neuronové sítě	23
3.2.1	Metoda podpůrných vektorů	23
3.2.2	Samoorganizace mapy	25
3.2.3	Konvoluční neuronové sítě	26
3.2.4	Autoenkodéry	27
3.3	Další vybrané metody využívající adaptivní filtry	28
3.3.1	Metoda set-membership	28
3.3.2	Algoritmus Learning Entropy	29

3.3.3	Algoritmus Error and Learning Based Novelty Detection	30
4	Zobecněné Paretovo rozdělení	33
4.1	Metoda Peak-over-Threshold	35
4.2	Metody odhadu parametrů zobecněného Paretova rozdělení	35
4.2.1	Metoda maximální věrohodnosti	36
4.2.2	Metoda momentů	36
4.2.3	Kvazi-ML metoda	38
5	Algoritmus Extreme Seeking Entropy	39
6	Výsledky detekce novosti algoritmem Extreme Seeking Entropy	43
6.1	Chaotická časová řada Mackey-Glass a detekce pertubace	43
6.2	Detekce změny rozptylu šumu v náhodném datovém toku	44
6.3	Detekce skokové změny parametrů generátoru signálu	46
6.4	Detekce náhlé absence šumu	49
6.5	Detekce změny trendu	49
6.6	Detekce epilepsie v EEG záznamu myši	52
6.7	Vyhodnocení úspěšnosti detekce skokové změny parametrů generátoru signálu	54
6.8	Vyhodnocení úspěšnosti detekce skokové změny trendu	57
6.9	Evaluae ROC křivky pro detekci změny trendu	59
6.9.1	Popis experimentu	59
6.9.2	Konstrukce ROC křivky	60
6.9.3	Výsledky experimentu	61
6.10	Vyhodnocení výpočetní náročnosti metod odhadu parametrů zobecněného Pareto- tova rozdělení	65
6.10.1	Motivace	65
6.10.2	Specifikace experimentu	66
6.10.3	Výsledky a diskuze	67
7	Závěr	71
7.1	Možné směry budoucího výzkumu	72
	Publikace autora	73
	Literatura	75
	Příloha	
A	Vybrané publikace	

- B Případová studie použití algoritmu Learning Entropy a adaptivního fuzzy filtru pro detekci změn stavů bioprocesu
 - B.1 Popis bioprocesu a specifikace problému
 - B.2 Experiment a zhodnocení
- C Vyhodnocení přesnosti detekce změny trendu a skokové změny parametrů generátoru signálu

Seznam použitých zkratek

AISLE	approximate individual sample learning entropy
AUROC	area under the receiver operating characteristics
ELBND	error and learning based novelty detection
ESE	extreme seeking entropy
FIR	finite impulse response
GEV	generalized extreme value
GMM	Gaussian Mixture Model
GNGD	generalized normalized gradient descend
GPD	generalized Pareto distribution
HMM	hidden Markov model
HONU	higher order neural unit
IIR	infinite impulse response
LE	learning entropy
LNU	linear neural unit
LMS	least mean squares
MOM	method of moments
NLMS	normalized least mean squares
PCA	principal component analysis
POT	peak-over-threshold
QNU	quadratic neural unit
RLS	recursive least squares
ROC	receiver operating characteristics
SNR	signal-to-noise ratio
SM-NLMS	set-membership normalized least mean squares
SVM	support vector machine

Seznam symbolů

\mathbb{N}	množina přirozených čísel
\mathbb{R}	množina reálných čísel
k	diskrétní časový index
$e(k)$	chyba predikce
$v(k)$	aditivní šum
$x(k)$	vstupní signál
$y(k)$	hodnota měřeného signálu (časové řady)
$\hat{y}(k)$	výstup adaptivního filtru
μ	rychlost učení
n_w	počet adaptivních parametrů
w_i	i -tý adaptivní parametr
\mathbf{w}	vektor adaptivních parametrů
$\text{col}\mathbf{x}$	uspořádaný sloupcový vektor vstupů
Ru^l	l -té pravidlo fuzzy systému
A_i^j	fuzzy množina ve vstupním prostoru
B^j	fuzzy množina ve výstupním prostoru
μ_B	funkce příslušnosti k fuzzy množině B
J	kriteriální funkce
E	střední hodnota
σ_n	směrodatná odchylka šumu
Δ	diference
∇	operátor nabla
R_{xx}	autokorelační matice
R_x	výběrová kovarianční matice
$\overline{b^j}$	střed fuzzy množiny ve výstupním prostoru
$\overline{x_i^j}$	střed fuzzy množiny ve vstupním prostoru
σ_i^j	parametr Gaussovske fuzzy množiny
q_{max}	maximální počet iterací algoritmu gradient descent

$U(a, b)$	rovnorné rozložení
$N(0, 1)$	normální rozdělení
α	uspořádaná množina prahů
n_α	počet prahů pro Learning Entropy
E_A	Approximate Individual Sample Learning Entropy
$LE(k)$	hodnota Learning Entropy
$ELBND(k)$	hodnota Error and Learninb Based Novelty Detection
F_u	distribuční funkce hodnot překračujících práh u
ξ_i	parametr tvaru i -tého zobecněného Paretova rozdělení
μ_i	parametr polohy i -tého zobecněného Paretova rozdělení
σ_i	parametr měřítka i -tého zobecněného Paretova rozdělení
$f_{\xi, \mu, \sigma}$	hustota pravděpodobnosti zobecněného Paretova rozdělení
$F_{\xi, \mu, \sigma}$	distribuční funkce zobecněného Paretova rozdělení
l	počet vzorků pro odhad parametrů zobecněného Paretova rozdělení

1 Úvod

Tato disertační práce je věnována problematice využití adaptivních systémů při analýze dat. Vzhledem k exponenciálnímu celosvětovému nárůstu objemu dat [1] a ke zvyšování jejich variability roste i potřeba tato data analyzovat, kategorizovat a vytěžovat. Analýzou dat rozumíme proces, kdy z nezpracovaných naměřených dat získáme nějakou interpretovatelnou informaci, s kterou pak lze dále pracovat. Jedna z možných důležitých interpretací nově získaných dat je, zda-li se nově získaná data nějakým zásadním způsobem odlišují od předchozích dat. Této problematice se věnuje obor detekce novosti, neboli anomálií, který spadá do oblasti vytěžování dat a strojového učení. Úspěšná detekce novosti pak může být využita k vícero účelům. Například k diagnostice sledovaného procesu, ke změně struktury nebo parametrů adaptivního modelu za účelem zlepšení predikce, z konkrétních aplikací pak k odhalení neoprávněného vniknutí do sítě nebo zneužití dat, v lékařství se detekce novosti používá k diagnostickým účelům, z průmyslových aplikací pak k detekci poruchy a monitoringu stavu strojů, senzorů, ve zpracování textových dat k detekci nových témat, originalnosti textů atd. Spektrum využití je velice široké.

V oblasti detekce novosti byla v posledních desetiletích intenzivního vývoje navrhována celá řada algoritmů. Vzhledem k rostoucímu výpočetnímu výkonu a rozmanitosti analyzovaných dat rostla i potřeba nových algoritmů. Nové algoritmy typicky předčily ostatní v rámci jedné aplikace, respektive v rámci jednoho typu dat. Doposud se však nepodařilo vytvořit algoritmus, který by ve všech (nebo alespoň ve významné části) oblastech použití předčil již publikované algoritmy. I proto vznikají v oblasti detekce novosti neustále nové přístupy, které navíc umožňují analyzovat nové typy dat.

1.1 Cíle dizertační práce

Cíle předkládané dizertační práce jsou:

1) Návrh algoritmu pro detekci novosti v datech s využitím adaptivních systémů

Navrhovaný algoritmus pro detekci novosti bude využívat adaptivní systémy a bude mít interpretovatelný výstup. Jeho použití by mělo být možné v kombinaci libovolným

adaptivním systémem. Algoritmus bude implementován v jazyce Python, který je v současné době jedním z nejrozšířenějších programovacích jazyků.

2) Otestování navrženého algoritmu na syntetických datech

Navržený algoritmus bude otestován na syntetických datech, která budou simulovat různé typy novosti a budou obsahovat šum. Mezi testovanými daty budou i data časových řad obsahujících trend. Výsledky algoritmu budou porovnány se obdobnými soudobými metodami využívajícími adaptivní systémy, konkrétně s algoritmem Learning Entropy a Error and Learning Based Novelty Detection.

3) Provedení případových studií na reálných datech z oblasti biomedicíny a chemie

Pro otestování navrženého algoritmu je zásadní provedení případových studií na reálných datech.

4) Vyhodnocení kvality navrženého algoritmu z pohledu úspěšnosti detekce novosti

Pro vyhodnocení kvality navrženého algoritmu budou určeny dosažené přesnosti detekce v různých scénářích. Dále budou pro vybrané scénáře určeny ROC křivky a vyhodnocena plocha pod nimi. Vyhodnocení bude provedeno pro scénáře s různými poměry signál-šum. Výsledky budou porovnány s výsledky algoritmů Learning Entropy a Error and Learning Based Novelty Detection.

1.2 Členění práce

Předkládaná disertační práce je členěna do sedmi kapitol, přičemž kapitoly 2, 3, 4 jsou rešeršního charakteru. Druhá kapitola obsahuje přehled adaptivních filtrů a metod, které byly v rámci práce použity. Třetí kapitola je věnována přehledu různých metod detekce novosti a obsahuje i oblasti jejich využití. Čtvrtá kapitola je věnována zobecněnému Paretovu rozdělení, které bylo použito v navrženém algoritmu detekce novosti.

Těžiště práce je v kapitolách 5 a 6, které jsou autorovým příspěvkem k řešení problematiky detekce novosti s využitím adaptivních systémů. Pátá kapitola obsahuje popis nově navrženého algoritmu nazvaném Extreme Seeking Entropy a krátce jsou zmíněny i jeho omezení. V šesté kapitole jsou výsledky tohoto algoritmu v porovnání s dalšími vybranými metodami adaptivní detekce novosti. Je zde také ukázána přesnost jeho detekce v různých scénářích a experimentální vyhodnocení času potřebného k výpočtu odhadu parametrů zobecněného Paretova rozdělení. Sedmá kapitola je závěrečnou a shrnuje dosažené výsledky.

2 Číslicové adaptivní filtry a algoritmy

Tato kapitola je věnována diskretním adaptivním filtrům, které byly v průběhu práce na tématu dizertační práce použity (viz podkapitola 2.1) a algoritmům, které byly k jejich adaptaci použity (viz podkapitola 2.2). Z pohledu klasifikace filtrů dle impulsní charakteristiky se jedná o filtry s konečnou impulsní charakteristikou i o filtry s nekonečnou impulsní charakteristikou. Z pohledu lineární závislosti adaptabilních parametrů potom na filtry lineární a nelineární v parametrech.

Obecně problém filtrace spočívá ve zpracování signálu filtrem tak, že ze signálu získáme nějakou užitečnou informaci [2].

Pro úplnost uveďme, že signálem rozumíme fyzikální veličinu, která se mění v čase, prostoru nebo v jakékoliv jiné nezávislé proměnné (obecně proměnných) [3]. V rámci dizertace jsou analyzovány pouze signály, které se mění v čase.

Jedno ze základních dělení signálů je rozdělení na signály spojité a diskretní, přičemž signál může být spojitý, resp. diskretní v čase nebo amplitudě. Uvažujme signál spojitý v čase i amplitudě $s(t)$. Pro takový signál platí, že hodnota jeho amplitudy $s(t) \in \mathbb{R}$ a hodnota nezávislé proměnné t je z nějakého intervalu $(t_1; t_2)$, kde $t_1 \in \mathbb{R}$ a $t_2 \in \mathbb{R}$ (předkládaná dizertační práce se nezabývá komplexními signály). Signál spojitý v amplitudě a diskretní v čase vznikne vzorkováním původního signálu $s(t)$ pomocí vzorkovací funkce ve tvaru

$$v_v(t) = \sum_{k=-\infty}^{\infty} \delta(t - k\Delta t) \quad (2.1)$$

kde Δt je vzorkovací perioda (uvažujeme konstantní vzorkovací periodu) a δ je, z pohledu teorie distribucí, lineární funkcionál na prostoru testovacích funkcí φ (všech hladkých funkcí na \mathbb{R} s kompaktním nosičem, které mají požadovaný počet derivací) definovaný jako

$$\langle \delta, \varphi \rangle = \varphi(0) \quad (2.2)$$

pro každou testovací funkci φ [4]. Pro navzorkovaný signál tedy platí

$$s(k) = s(t) \cdot v_v(t) \quad (2.3)$$

a výsledkem je posloupnost vzorků $s(k)$. Signál spojitý v čase a diskretní v amplitudě získáme aplikací kvantizační funkce $Q(s)$, která převede hodnotu signálu na číslo z nějaké množiny přípustných hodnot (typicky to bývá celé číslo, případně číslo s plovoucí desetinnou čárkou). Existuje celá řada kvantizačních algoritmů (více viz [5]), takže pro úplnost uvedeme pouze základní uniformní kvantizátor s velikostí kvantizačního kroku definovanou jako

$$q = \frac{s_{max} - s_{min}}{L} \quad (2.4)$$

kde L určuje počet intervalů o délce q . Potom kvantizovanou hodnotu můžeme určit jako

$$Q(s) = \left\lfloor s - \frac{s_{min}}{q} \right\rfloor q + \frac{q}{2} + s_{min} \quad (2.5)$$

Signál, který je diskretní v čase i amplitudě bývá označován jako digitální signál a právě těmito signály se předložená dizertační práce zabývá.

Signály dále můžeme rozdělit na skalární a vektorové. Vektorovým (někdy též označovaným jako vícekanálový) signálem je například EEG. Některé systémy pro měření EEG využívají až 256 kanálů [6]. Takový signál můžeme v časovém okamžiku k reprezentovat 256-ti dimenzionálním vektorem

$$\mathbf{s}(k) = [s_1(k), \dots, s_{256}(k)]^T \quad (2.6)$$

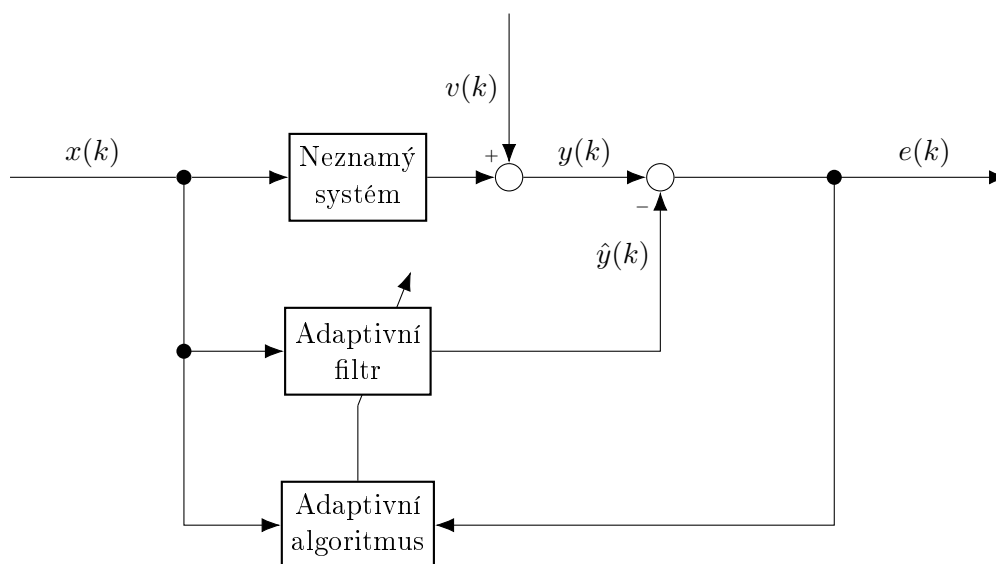
kde i -tá složka $s_i(k)$ odpovídá signálu i -tého kanálu. Skalární signál (někdy též označovaný jako jednorozměrný) je pak např. právě i -tá složka signálu $\mathbf{s}(k)$.

Z pohledu počtu nezávislých proměnných, jejichž funkcí lze signál vyjádřit, lze rozlišovat mezi signály jednorozměrnými a vícerozměrnými. Jednorozměrným signálem je například záznam jednoho kanálu EEG, kde hodnota signálu $s_i(k)$ je závislá pouze na čase. Vícerozměrným signálem může např. digitální obraz, jehož intenzita je funkcí souřadnic $I(x, y)$.

Další dělení signálu je na signály deterministické a náhodné. Pokud můžeme popsat signál pomocí explicitního matematického zápisu, tabulky nebo pravidlem tak, že pro každý časový okamžik získáme jednoznačnou hodnotu, jedná se o signál deterministický. V opačném případě jsou signály, jejichž hodnota lze určit pouze s nějakou pravděpodobností, označovány jako náhodné (stochastické). Typickým příkladem náhodného signálu je např. bílý šum. Ze stochastických signálů je, z pohledu této práce, zajímavé ještě dělení na ergodické a stacionární signály. Pro stacionární signály je jejich pravděpodobnostní popis nezávislý na počátku časové osy. Ergodické signály jsou pak signály jejichž statistické vlastnosti lze určit z libovolné, dostatečně velké, množiny dat a tyto vlastnosti se v čase nemění.

Filtr můžeme použít v následujících základních úlohách zpracování signálů:

- filtrace
- vyhlazování
- predikce



Obrázek 2.1: Schéma adaptivní filtrace

V rámci této práce byly všechny filtry využity k predikci, jejíž chyba e byla využita k adaptaci parametrů daného filtru (více viz podkapitola 2.2).

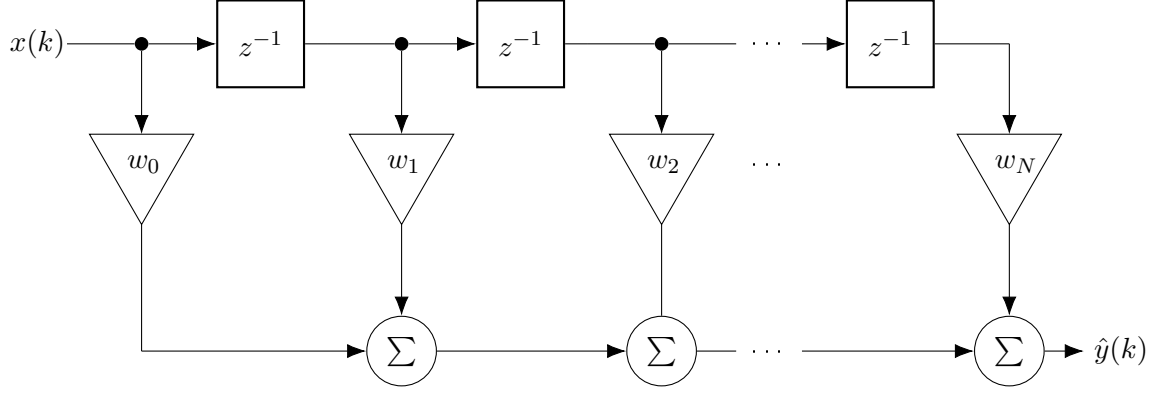
2.1 Adaptivní filtry

V této podkapitole jsou stručně popsány adaptivní filtry, které byly v rámci práce použity. Adaptivní filtry s konečnou odezvou (FIR - finite impulse response) jsou popsány v podkapitole 2.1.1, Volterrovy filtry v podkapitole 2.1.2, polynomiální neuronové jednotky v kapitole 2.1.3 a adaptivní fuzzy filtry v podkapitole 2.1.4. Dále jsou zde popsány vybrané adaptivní algoritmy, jmenovitě LMS (least mean squares) a NLMS (normalized least mean squares) v podkapitole 2.2.1, RLS (recursive least squares) v podkapitole 2.2.2, GNGD (generalized normalized gradient descent) v podkapitole 2.2.3 a varianta algoritmu gradient descent pro fuzzy filtry v podkapitole 2.2.4.

Schéma znázorňující základní problém adaptivní filtrace je znázorněno na obrázku 2.1. Značení v uvedeném schématu má následující významy: $x(k)$ je vstupní signál, $v(k)$ je aditivní šum přidaný k výstupu neznámého systému, $y(k)$ je měřený výstup systému, $\hat{y}(k)$ je

výstup adaptivního filtru a $e(k)$ je chyba filtru (nebo také chyba predikce). Schéma může být modifikováno např. tak, že vstupem adaptivního filtru je historie výstupů neznámého systému.

2.1.1 Lineární FIR filtry



Obrázek 2.2: Blokový diagram FIR filtru

Výstup FIR (finite-impulse-response) filtru $\hat{y} \in \mathbb{R}$ v diskrétním čase $k \in Z$ je popsán rovnicí

$$\hat{y}(k) = \sum_{i=0}^N w_i \cdot x(k-i) \quad (2.7)$$

kde $w_i \in \mathbb{R}$ hodnota i -tého koeficientu, $x(k-i)$ je hodnota vstupu $x \in \mathbb{R}$ posunutá o i hodnot v čase, hodnota N je řád filtru. Pokud jsou koeficienty FIR filtru v čase adaptovány, přejde rovnice (2.7) do tvaru

$$\hat{y}(k) = \sum_{i=0}^N w_i(k) \cdot x(k-i) \quad (2.8)$$

kde člen $w_i(k)$ reprezentuje hodnoty koeficientů filtru v diskrétním čase k . V následujícím textu, již uvažujeme, že veškeré adaptivní filtry mají v čase proměnné koeficienty w . Někdy se pro popis výstupu FIR filtru využívá operátoru konvoluce, potom pro výstup FIR filtru platí

$$\hat{y}(k) = h(k) * x(k) = x(k) * h(k) \quad (2.9)$$

Hodnoty $h(k)$ v reprezentují impulzní odezvu filtru. Impulzní odezva filtru je definovaná jako

$$h(k) = \sum_{i=0}^N w_i \cdot \delta(k-i) = \begin{cases} w_i & 0 \leq k \leq N \\ 0 & k < 0 \vee k > N \end{cases} \quad (2.10)$$

Konečnost impulzní odezvy je dána konečným počtem n_w koeficientů filtru, přičemž pro počet koeficientů filtru platí vztah $n_w = N + 1$. Pokud koeficienty filtru splňují podmínku $\forall i : w_i <$

∞ , potom je FIR filtr stabilní.

Uvedný filtr (viz rovnice (2.7)) lze zapsat ve tvaru

$$\hat{y}(k) = \mathbf{w}(k) \cdot \mathbf{x}(k) \quad (2.11)$$

kde $\mathbf{w}(k)$ je vektor vah

$$\mathbf{w}(k) = [w_1, \dots, w_N] \quad (2.12)$$

a $\mathbf{x}(k)$ je vstupní vektor

$$\mathbf{x}^T(k) = [x(k), x(k-1), \dots, x(k-N)]. \quad (2.13)$$

Blokové schéma zobrazující funkci FIR filtru je zobrazeno na obrázku 2.2. Bloky označené z^{-1} reprezentují časové zpoždění, respektive operátor z^{-1} (využijeme-li notaci Z-transformace).

2.1.2 Volterrový filtry

Jeden z filtrů, který se používá k modelování nelineárních systémů je nelineární Volterrovův filtr (za určitých okolností je ekvivalentní s tzv. Higher Order Neural Units - HONU [7]). Výstup $\hat{y}(k)$ těchto filtrů, využívajících zkrácených Volterrových řad, je popsán rovnicí

$$\begin{aligned} \hat{y}(k) = w_0 + \sum_{i=0}^{N-1} w_1(i)x(k-i) + \sum_{i_1=0}^{N-1} \sum_{i_2=i_1}^{N-1} w_2(i_1, i_2)x(k-i_1)x(k-i_2) + \dots \\ \dots \sum_{i_1=0}^{N-1} \dots \sum_{i_p=i_{p-1}}^{N-1} w_p(i_1, \dots, i_p)x(k-i_1) \dots x(k-i_p) \end{aligned} \quad (2.14)$$

kde w_0 je konstanta, $\{w_j(i_1, \dots, i_j), 1 \leq j \leq p\}$ je množina koeficientů Volterrovo jader j -tého řádu a $x(k)$ je vstupní signál. Filtr pracuje s pamětí N vzorků, parametr p určuje řád filtru. Analogický zápis využívající vektorové násobení

$$\hat{y}(k) = \mathbf{w} \cdot \mathbf{colx} \quad (2.15)$$

kde \mathbf{w} je uspořádaný vektor koeficientů Volterrovo jader a \mathbf{colx} uspořádaný vektor ve tvaru

$$\begin{aligned} \mathbf{colx} = [1, x(0), \dots, x(N-1), x(1) \cdot x(2), \dots, x(1) \cdot x(N-1), x(2)^2, x(2) \cdot x(3), \dots, \\ x(2) \cdot x(N-1), \dots, x(N-2) \cdot x(N-1), x(N-1)^2, \dots, \dots, x(N-1)^p]^T \end{aligned} \quad (2.16)$$

Určitým faktorem limitujícím použití Volterrových filtrů je vysoký počet jejich parametrů, přičemž každé zvýšení počtu vzorků v paměti, nebo řádu filtru, výrazně zvýší počet jeho

parametrů. Pro počet parametrů n_W Volterrova filtru s délkou paměti N a řádu p je dán jako

$$n_w(p, N) = \frac{(N + p)!}{N!p!} \quad (2.17)$$

kde $N!$ je faktoriál N a $p!$ je faktoriál p . Např. pro $N = 3$ a $p = 3$ má filtr 20 parametrů, pro $N = 4$ a $p = 3$ má 35 parametrů, pro $N = 4$ a $p = 4$ již 70 parametrů. Pro zpracování signálů v reálném čase je tedy využití Volterrových filtrů s velkou pamětí a vysokým řádem, vzhledem k vysokému počtu parametrů, komplikované. Volterrovův filtr druhého řádu bývá někdy označován Quadratic neural unit (QNU) a třetího řádu Cubic neural unit (CNU) (více viz [7]). Volterrovův filtr prvního řádu je variantou standartního FIR filtru (viz kap. 2.1.1) v případě, že $w_0 = 0$. V případě, že $w_0 \neq 0$, je tento filtr typu IIR, tedy má nekonečnou impulsní charakteristiku.

2.1.3 Polynomiální neuronové jednotky

Jedním z typů nelineární filtrů, které jsou ale lineární v adaptivních parametrech jsou polynomiální neuronové jednotky, někdy též nazývané HONU (Higher Order Neural Units) [7]. Výstup HONU p -tého řádu je definován jako

$$\hat{y}(k) = \sum_{i_1=0}^n \sum_{i_2=i_1}^n \cdots \sum_{i_p=i_{p-1}}^n w_{i_1, i_2, \dots, i_p} x_{i_1} \cdot x_{i_2} \cdots x_{i_p} \quad (2.18)$$

přičemž člen $x_{0, \dots, 0} = 1$ je označován jako bias, w_{i_1, i_2, \dots, i_p} jsou váhy a x_{i_j} je j -tý vstup. Uvedený filtr lze reprezentovat pomocí násobení vektorů, obdobně jako Volterrovy filtry, jako

$$\hat{y}(k) = \mathbf{w} \cdot \mathbf{colx} \quad (2.19)$$

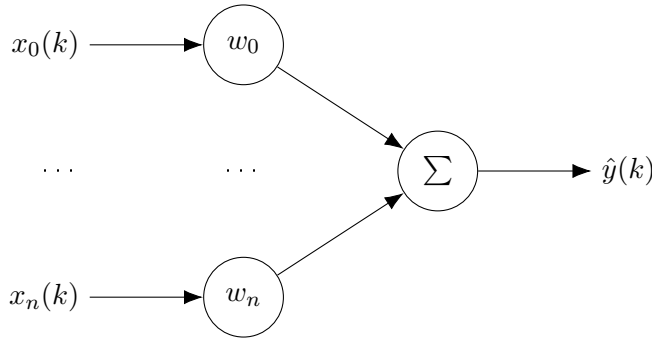
kde w je uspořádaný vektor adaptivních parametrů

$$\mathbf{w} = [w_{0, \dots, 0}, \dots, w_{n, \dots, n}] \quad (2.20)$$

a \mathbf{colx} je odpovídajícím způsobem uspořádaný vektor vstupů ve tvaru

$$\mathbf{colx} = [1, x_{0, \dots, 1}, \dots, x_{n, \dots, n}]^T \quad (2.21)$$

přičemž pokud vektor vstupů obsahuje časově zpožděné vzorky vstupního signálu, jsou HONU identické s Volterrovými filtry. Pokud je vstupem $n + 1$ -dimenzionální vektor různých vstupů, tak je HONU kombinačním filtrem. HONU prvního řádu bývá označována jako lineární neuronová jednotka (LNU). Pokud jsou vstupem do LNU časově posunuté hodnoty vstupního signálu, je identická s klasickým FIR filtrem (viz kapitola 2.1.1), protože její výstup je defi-



Obrázek 2.3: LNU jako lineární kombinační filtr

nován jako

$$\hat{y}(k) = \sum_{i=0}^{n_w} w_i \cdot x(k-i) \quad (2.22)$$

Pokud je vstupem n -dimenzionální vektor n různých vstupů, potom je LNU klasickým lineárním kombinačním filtrem (viz obr. 2.3). Použijeme-li terminologii neuronových sítí, potom je lineárním neuronem, kde \mathbf{w} je vektor synaptických vah, $\mathbf{x}(k)$ je vstupní vektor a přenosová funkce tohoto neuronu realizuje lineární kombinaci.

$$\hat{y}(k) = \sum_{i=0}^n w_i \cdot x_i(k) \quad (2.23)$$

Často používanými HONU jsou jednotky druhého (quadratic neural unit - QNU) a třetího řádu (cubic neural unit - CNU). Podobně jako pro Volterrovy filtry, i úskalím použití HONU vyššího řádu s velkou pamětí je velký počet parametrů.

2.1.4 Fuzzy filtry

Jedním z rozšířených typů nelineárních adaptivních filtrů jsou filtry založené na fuzzy logice [8]. V rámci práce byl použit fuzzy adaptivní filtr, tvořený Mamdaniho fuzzy systémem se součinným inferenčním mechanismem a defuzzifikací využívající metody těžiště typických hodnot. Báze pravidel tohoto fuzzy filtru je tvořena M pravidly, kdy l -té pravidlo je ve tvaru

$$Ru^l : IF x_1 is A_1^l and x_2 is A_2^l and \dots x_n is A_n^l THEN \hat{y} is B^l \quad (2.24)$$

kde A_i^l je i -tá množina ve vstupním prostoru $U \subset \mathbb{R}^n$, B^l je fuzzy množina ve výstupním prostoru $V \subset \mathbb{R}$ a $x_i \in U$ a $\hat{y} \in V$ jsou lingvistické proměnné. Fuzzy množiny ve výstupním prostoru jsou singletony, obsahují tedy jediný prvek x^* z univerza X a pro jejich funkce

příslušnosti (charakteristické funkce) platí, že

$$\mu_B(x^*) = 1 \quad (2.25)$$

takže jádro této množiny je identické s jejím nosičem a obsahují pouze jeden stejný prvek.

$$\ker(B) = \text{supp}(B) = \{1/x^*\} \quad (2.26)$$

Studovaný fuzzy systém využívá součinnové konjukce a Larsenovy implikace. Tím se převede vyhodnocení implikace na několik součinů tak, že pro j -té pravidlo platí

$$\begin{aligned} [\mu_{A_1^j}(x_1) \text{ AND } \mu_{A_2^j}(x_2) \text{ AND } \dots \text{ AND } \mu_{A_n^j}(x_n)] &\implies \mu_{B^j}(\hat{y}) = \\ &= \mu_{A_1^j}(x_1) \cdot \mu_{A_2^j}(x_2) \cdot \dots \cdot \mu_{A_n^j}(x_n) \cdot \mu_{B^j}(\hat{y}) \end{aligned} \quad (2.27)$$

kde $\mu_{A_i^j}(x_i)$ je funkce příslušnosti k i -té množině j -tého pravidla ve vstupním prostoru a μ_{B^j} je funkce příslušnosti ke množině ve výstupním prostoru.

Defuzzifikace převádí fuzzy do její reprezentace pomocí jediného čísla z množiny ostrých hodnot. Metodou těžiště typických hodnot se výstup fuzzy systému určí podle rovnice

$$y^* = \frac{\sum_{j=1}^M \bar{b}^j z_j}{\sum_{j=1}^M z_j} \quad (2.28)$$

kde \bar{b}^j je střed j -té fuzzy množiny reprezentující příslušné pravidlo a z_j je její váha ve smyslu hodnoty funkce příslušnosti tohoto pravidla.

Výstup uvedeného fuzzy systému je potom ve tvaru

$$\hat{y}(\mathbf{x}) = \frac{\sum_{j=1}^M \bar{b}^j [\prod_{i=1}^n \mu_i^j(x_i)]}{\sum_{j=1}^M [\prod_{i=1}^n \mu_i^j(x_i)]} \quad (2.29)$$

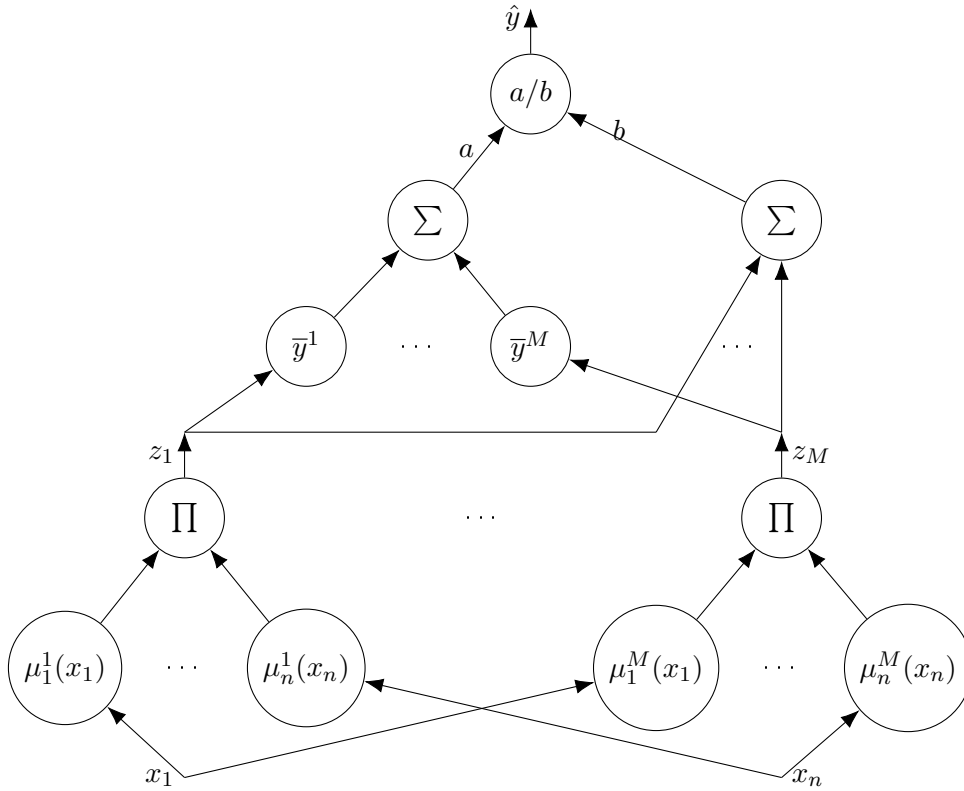
přičemž \mathbf{x} je vektor vstupů o délce n

$$\mathbf{x} = [x_1, \dots, x_n] \quad (2.30)$$

a \bar{b}^j je střed výstupní množiny B^j , což je fuzzy množina j -tého pravidla. Funkce filtru lze znázornit třívrstvou dopřednou sítí (viz obrázek 2.4).

V první vrstvě jsou určeny váhy jednotlivých pravidel, tedy jsou vypočteny hodnoty z_j , kde $j = 1, \dots, M$. V druhé vrstvě jsou váhy vynásobeny polohami středů množin ve výstupním prostoru a sečteny (výpočet a), a také jsou váhy jednotlivých pravidel sečteny (výpočet b). Ve třetí vrstvě se pak vypočte výstup fuzzy systému jako $\hat{y} = \frac{a}{b}$ (více viz podkapitola 2.2.4).

Nespornou výhodou uvedeného fuzzy adaptivního filtru je, že tvoří tzv. univerzální apro-



Obrázek 2.4: Fuzzy filtr jako dopředná síť

ximátor. Při vhodně zvolené bázi pravidel a typu funkcí příslušnosti množin ve vstupním prostoru tak dokáže aproximovat libovolnou funkci s libovolně velkou přesností (více viz. [9]).

2.2 Adaptivní algoritmy

V této podkapitole jsou popsány adaptivní algoritmy, které byly v rámci dizertační práce vyzkoušeny. Jedná se o LMS a NLMS (viz podkap. 2.2.1), Generalized Normalized Gradient Descent (GNGD, viz podkap. 2.2.3) a algoritmus Gradient descent ve verzi pro adaptivní fuzzy filtry s Gausovskými funkcemi příslušnosti ve vstupním prostoru (viz podkap. 2.2.4).

2.2.1 Algoritmy LMS a NLMS

Při použití LMS algoritmu (LMS - least mean squares [2]) optimální hodnoty adaptivních parametrů filtru $\mathbf{w}(k+1)$ minimalizují střední kvadratickou chybu predikce $J(k)$ definovanou jako

$$J(k) = E[|e(k)|^2] \quad (2.31)$$

kde $E[\cdot]$ značí střední (očekávanou) hodnotu, a $e(k)$ je chyba predikce, definovaná jako rozdíl výstupu adaptivního filtru a skutečné hodnoty

$$e(k) = y(k) - \hat{y}(k). \quad (2.32)$$

Hodnoty adaptivních parametrů filtru jsou nalezeny gradientním algoritmem. S každými nově získanými daty jsou potom váhy filtru upraveny podle rovnice

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \Delta \mathbf{w}(k) \quad (2.33)$$

přičemž pro LMS algoritmus je přírůstek vah definován jako

$$\Delta \mathbf{w}(k) = -\frac{\mu}{2} \nabla_{\mathbf{w}} E[|e(k)|^2] = \mu E[\mathbf{x}(k)e(k)] \quad (2.34)$$

kde μ je rychlost učení (velikost kroku) ovlivňující rychlost konvergence algoritmu a ∇ je operátor nabra

$$\nabla = \left(\frac{\partial}{\partial w_1}, \dots, \frac{\partial}{\partial w_n} \right). \quad (2.35)$$

Protože jsou parametry filtru přepočítány s každými nově získanými daty (online), je možné nahradit očekávanou hodnotu $E[\mathbf{x}(k)e(k)]$ hodnotou okamžitou. Výpočet nových hodnot parametrů adaptivního filtru tak přejde do tvaru

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu \mathbf{x}(k)e(k) \quad (2.36)$$

přičemž pro konvergenci a stabilitu algoritmu musí být splněna podmínka pro velikost rychlosti učení

$$0 < \mu < \frac{2}{\lambda_{max}} \quad (2.37)$$

kde λ_{max} je největší vlastní číslo autokorelační matice

$$R_{xx} = E[\mathbf{x}(k)\mathbf{x}^T(k)]. \quad (2.38)$$

Pokud není podmínka splněna, pak je algoritmus nestabilní a hodnoty $\mathbf{w}(k)$ divergují. Pokud je naopak velikost rychlosti učení μ příliš malá, váhy konvergují pomalu.

Algoritmus NLMS (normalized least mean squares - NLMS [2]) řeší problém klasického LMS algoritmu, který v případě nevhodně škálovaného vstupu $\mathbf{x}(k)$ ztrácí stabilitu. Tento problém je vyřešen normalizací vstupu. Velikost přírůstku vah je tedy

$$\Delta \mathbf{w}(k) = \mu \frac{e(k)\mathbf{x}(k)}{\mathbf{x}^T(k)\mathbf{x}(k)}. \quad (2.39)$$

Podmínka pro velikost rychlosti učení zajišťující stabilitu algoritmu je v případě, že vstupní signál $x(k)$ není korelovaný s aditivním šumem

$$0 < \mu < 2. \quad (2.40)$$

Velikost optimální rychlosti učení je ovlivněná vlastnostmi aditivního šumu $n(k)$ a v případě, že tento šum není korelovaný se vstupním signálem \mathbf{x} je dána jako

$$\mu_{optimal} = \frac{E[|y(k) - \hat{y}(k)|^2]}{E[|e(k)|^2]} \quad (2.41)$$

Problém nastane v případě, že vektor $\mathbf{x}(k)$ je nulový. Z tohoto důvodu, se přidává do jmenovatele v rovnici 2.39 malá pozitivní konstanta $\epsilon > 0$, která řeší problém s potenciálním dělením nulou. Změna adaptivních vah filtru je v tomto případě dána jako

$$\Delta \mathbf{w}(k) = \mu \frac{e(k)\mathbf{x}(k)}{\mathbf{x}^T(k)\mathbf{x}(k) + \epsilon} \quad (2.42)$$

a podmínka pro velikost rychlosti učení přejde do tvaru

$$0 < \mu < 2 + \frac{\epsilon}{\mathbf{x}^T(k)\mathbf{x}(k)}. \quad (2.43)$$

2.2.2 Algoritmus RLS

Při použití rekurentní metody nejmenších čtverců s exponenciálním zapomináním (RLS - recursive least squares [2]) je odhad parametrů filtru získán na základě minimalizace kritériální funkce ve tvaru

$$J(k) = \sum_{j=j_1}^k \lambda^{k-j} e(j)^2 \quad (2.44)$$

která je exponenciálně váženým součtem chyb výstupu posledních $k - j_1$ vzorků, přičemž parametr $\lambda \in (0; 1)$ je tzv. faktor exponenciálního zapominání. Chyba výstupu je pak definována jako

$$e(j) = y(j) - \hat{y}(j) \quad (2.45)$$

kde

$$\hat{y}(j) = \mathbf{w}^T(k)\mathbf{x}(j) \quad (2.46)$$

přičemž $\mathbf{w}^T(k)$ je vektor adaptivních parametrů filtru

$$\mathbf{w}^T(k) = [w_0(k), w_1(k), \dots, w_n(k)] \quad (2.47)$$

a $\mathbf{x}(k)$ je vstupní vektor v diskrétním časovém okamžiku k obsahující posledních $n + 1$ vzorků a definovaný jako

$$\mathbf{x}(j) = [x(j), x(j-1), \dots, x(j-n)]^T \quad (2.48)$$

kde parametr n se označuje jako řád filtru. Z uvedené kriteriální funkce 2.44 je tedy zřejmé, že historicky starší chyby výstupu mají exponenciálně klesající význam a vzorky, které jsou starší než $k - j_i$ vzorků, nejsou pro odhad parametrů filtru použity. Pro hodnoty adaptivních parametrů, které minimalizují výše uvedenou kriteriální funkci (2.44) v diskrétním časovém okamžiku k pak platí

$$\mathbf{w}(k) = \mathbf{w}(k-1) + \mathbf{P}(k)\mathbf{x}(k)[d(k) - \mathbf{x}^T(k)\mathbf{w}(k-1)] \quad (2.49)$$

kde matice $\mathbf{P}(k)$ je

$$\mathbf{P}(k) = \frac{1}{\lambda} \left[\mathbf{P}(k-1) - \frac{\mathbf{P}(k-1)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{P}(k-1)}{\lambda + \mathbf{x}^T(k)\mathbf{P}(k-1)\mathbf{x}(k)} \right]. \quad (2.50)$$

Člen $\mathbf{x}^T(k)\mathbf{w}(k-1)$ v rovnici (2.49) reprezentuje apriorní chybu filtru, která je vypočtena ještě před korekcí adaptivních vah¹. Dále poznamenejme, že $\mathbf{P}(k)$ je inverzní maticí k vážené výběrové kovarianční matici $\mathbf{R}_x(k)$ definované jako

$$\mathbf{R}_x(k) = \lambda^k \mathbf{R}_x(0) + \sum_{j=1}^k \lambda^{k-j} \mathbf{x}(j)\mathbf{x}^T(j) = \lambda \mathbf{R}_x(k-1) + \mathbf{x}(k)\mathbf{x}^T(k) \quad (2.51)$$

přičemž $\mathbf{R}_x(0)$ je počáteční hodnota. V praxi se počáteční hodnota matice $\mathbf{P}(0) = \mathbf{R}_x^{-1}(0)$ volí jako

$$\mathbf{P}(0) = \delta \cdot \mathbf{I} \quad (2.52)$$

kde δ je dostatečně velká pozitivní konstanta a \mathbf{I} je jednotková matice (v některé literatuře nazývaná jako matice identity). Pro signály s vysokým poměrem výkon-šum se volí malé hodnoty δ , pro signály s malým poměrem výkon-šum pak velké hodnoty δ . Pokud je k dispozici apriorní informace o σ_x^2 tedy varianci vstupního signálu $x(k)$, volí se hodnota konstanty podle [11] jako

$$\delta > 100\sigma_x^2. \quad (2.53)$$

Počáteční hodnota adaptivních parametrů se obvykle volí jako

$$\mathbf{w}(0) = 0. \quad (2.54)$$

¹Pokud uvažujeme aposteriorní chybu, kterou vypočteme po korekci vah, tak se nabízí určitá podobnost s Kalmanovým filtrem. Více o souvislosti mezi LMS algoritmy a Kalmanovým filtrem viz [10]

2.2.3 Algoritmus Generalized Normalized Gradient Descent

Algoritmus GNGD (Generalized Normalized Gradient Descent - více viz [2]) řeší problém případné pomalé konvergence algoritmu NLMS zavedením dalšího kompenzačního členu, který ovlivňuje velikost kroku při gradientní adaptaci. Nejprve uvažujme kvadratickou kritériální funkci ve tvaru

$$J(k) = \frac{1}{2}e^2(k) \quad (2.55)$$

a adaptaci parametrů ve tvaru (2.42). Malou pozitivní konstantu ϵ nahradíme dalším adaptivním členem

$$\epsilon(k+1) = \epsilon(k) - \rho \nabla_{e(k-1)} J(k) \quad (2.56)$$

přičemž

$$\begin{aligned} \frac{\partial J(k)}{\partial \epsilon(k-1)} &= \frac{\partial J(k)}{\partial e(k)} \frac{\partial e(k)}{\partial y(k)} \frac{\partial y(k)}{\partial \mathbf{x}(k)} \frac{\partial \mathbf{w}(k)}{\partial \eta(k-1)} \frac{\partial \eta(k-1)}{\partial \epsilon(k-1)} = \\ &= \frac{e(k)e(k-1)\mathbf{x}^T(k)\mathbf{x}(k-1)}{(\mathbf{x}^T(k)\mathbf{x}(k) + \epsilon(k-1))^2}. \end{aligned} \quad (2.57)$$

Adaptace parametrů je tedy dána jako

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \eta(k)e(k)\mathbf{x}(k) \quad (2.58)$$

kde

$$\eta(k) = \frac{\mu}{\mathbf{x}^T(k)\mathbf{x}(k) + \epsilon(k)} \quad (2.59)$$

$$\epsilon(k) = \epsilon(k-1) - \rho\mu \frac{e(k)e(k-1)\mathbf{x}^T(k)\mathbf{x}(k-1)}{(\mathbf{x}^T(k)\mathbf{x}(k) + \epsilon(k-1))^2} \quad (2.60)$$

přičemž parametr ρ je parametr adaptace velikosti kroku při spuštění algoritmu. Algoritmus GNGD je vhodný pro zpracování nelineárních a nestacionárních signálů.

2.2.4 Gradient descent pro fuzzy filtry

Při použití metody gradient descent pro adaptivní fuzzy filtry je potřeba nejdříve specifikovat strukturu filtru, tedy počet pravidel, množiny ve vstupním a výstupním prostoru, typ inferenční metody, fuzzifikace a defuzzifikace [9].

Uvažujme fuzzy systém specifikovaný v podkapitole 2.1.4, v jehož vstupním prostoru $U \subset R^n$ jsou Gaussovske funkce příslušnosti ve tvaru

$$\mu_{A_i^j}(x_i) = \exp \left[- \left(\frac{x_i - \bar{x}_i^j}{\sigma_i^j} \right)^2 \right] \quad (2.61)$$

kde \bar{x}_i^j je středem i -té vstupní množiny j -tého pravidla a σ_i^j je parametr, určující tvar, respektive šířku, této fuzzy množiny. Zobrazení, realizované výše specifikovaným fuzzy systémem je dáno rovnicí

$$\hat{y}(x) = \frac{\sum_{j=1}^M \bar{b}^j \left[\prod_{i=1}^n \exp\left(-\left(\frac{x_i - \bar{x}_i^j}{\sigma_i^j}\right)\right) \right]}{\sum_{j=1}^M \left[\prod_{i=1}^n \exp\left(-\left(\frac{x_i - \bar{x}_i^j}{\sigma_i^j}\right)\right) \right]} \quad (2.62)$$

přičemž parametr M určující počet množin je vzhledem k adaptaci parametrem fixním a parametry \bar{b}^j , σ_i^j a \bar{x}_i^j jsou parametry, které se adaptují. Dále uvažujme množinu N dvojic vstup-výstup, kde N odpovídá počtu vzorků experimentu.

$$\{\mathbf{x}(k), y(k)\}, \quad k = 1, 2, \dots, N \quad (2.63)$$

Při použití algoritmu gradient descent pro fuzzy filtr v kontextu detekce novosti se vždy s nově naměřenými daty tento filtr adaptuje. Adaptace probíhá s každými daty na základě minimalizace kritériální funkce

$$J(k) = \frac{1}{2} [\hat{y}(\mathbf{x}(k)) - y(k)]^2 \quad (2.64)$$

která je zvolená tak, aby měla právě jeden globální extrém, přičemž parametry jsou adaptovány dokud není dosaženo dostatečně malé chyby, nebo dokud není překročen předem stanovený maximální počet iterací q_{max} . Volba Gaussovských funkcí příslušnosti ve vstupním prostoru je výhodná z hlediska výpočtu derivace podle adaptabilních parametrů, neboť tato derivace existuje v každém bodě. Pro minimalizaci kritériální funkce (viz rovnice 2.64) použitím gradientní metody pak dostáváme vztahy pro hodnoty adaptivních parametrů v q -té iteraci jako

$$\bar{b}^j(q+1) = \bar{b}^j(q) - \mu \frac{\partial e}{\partial \bar{b}^j} \Big|_q = \bar{b}^j(q) - \mu (\hat{y} - y) \frac{\partial \hat{y}}{\partial a} \frac{\partial a}{\partial \bar{b}^j} = \bar{b}^j(q) - \mu \frac{\hat{y} - y}{b(q)} z^j(q) \quad (2.65)$$

a obdobně pro hodnoty středů vstupních množin \bar{x}_i^j a jejich příslušných parametrů σ_i^j dostaneme

$$\bar{x}_i^j(q+1) = \bar{x}_i^j(q) - \mu \frac{\partial e}{\partial \bar{x}_i^j} \Big|_q = \bar{x}_i^j(q) - \mu \frac{\hat{y} - y}{b(q)} [\bar{b}^j(q) - \hat{y}] z^j(q) \frac{2[x_i - \bar{x}_i^j(q)]}{\sigma_i^{j2}(q)} \quad (2.66)$$

$$\begin{aligned} \sigma_i^j(q+1) &= \sigma_i^j(q) - \mu \frac{\partial e}{\partial \sigma_i^j} \Big|_q = \\ &= \sigma_i^j(q) - \mu \frac{\hat{y} - y}{b(q)} [\bar{b}^j(q) - \hat{y}] z^j(q) \frac{2[x_i - \bar{x}_i^j(q)]^2}{\sigma_i^{j3}(q)} \end{aligned} \quad (2.67)$$

S každými nově získanými daty (v diskretním časovém okamžiku k) jsou tedy v q -té iteraci hodnoty parametru \bar{b}^j vypočítány podle rovnice (2.65), parametru \bar{x}_i^j podle rovnice (2.66) a parametru σ_i^j podle rovnice (2.67). Parametr μ je fixní a určuje velikost kroku. Algoritmus lze shrnout následujícími 5-ti kroky:

1. **Krok:** Určení počtu pravidel a počáteční nastavení parametrů $\bar{b}^j(0)$, $\bar{x}_i^j(0)$, $\sigma_i^j(0)$ a velikosti kroku μ .
2. **Krok:** Pro k -tou dvojici $(\mathbf{x}(k), y(k))$ v q -té iteraci jsou vypočteny hodnoty výstupních vrstev fuzzy systému (viz obr 2.4) podle rovnic (2.68), (2.69), (2.70) a (2.71).
3. **Krok:** Výpočet nových hodnot parametrů $\bar{b}^j(q+1)$, $\bar{x}_i^j(q+1)$, $\sigma_i^j(q+1)$ dle rovnic (2.65), (2.66), (2.67).
4. **Krok:** $q = q + 1$ a opakování kroků 2. a 3. pro , dokud není dosaženo maximálního množství iterací q_{max} nebo požadované přesnosti ϵ .
5. **Krok:** Návrat do kroku 2. pro hodnotu $k = k + 1$, tedy s novou dvojicí dat (\mathbf{x}, y) .

Rovnice popisující výstup jednotlivých vrstev fuzzy systému (viz obr. 2.4) následují.

$$z^j = \prod_{i=1}^n \exp \left[- \left(\frac{x_i(k) - \bar{x}_i^j(q)}{\sigma_i^j(q)} \right) \right] \quad (2.68)$$

$$b = \sum_{j=1}^M z^j \quad (2.69)$$

$$a = \sum_{j=1}^M \bar{b}^j(q) z^j \quad (2.70)$$

$$\hat{y} = \frac{a}{b} \quad (2.71)$$

Pro správnou funkci algoritmu je důležitá prvotní volba hodnot parametrů $\bar{b}^j(0)$, $\bar{x}_i^j(0)$, $\sigma_i^j(0)$. Náhodné hodnoty parametrů nejsou v případě použití tohoto algoritmu vhodné. Jedna z možných metod, jak vybrat hodnoty parametrů je využití prvních M dvojic vstup-výstup. Uvažujme množinu M dvojic vstup-výstup

$$\{\mathbf{x}(j), d(j)\}, j = 1, 2, \dots, M \quad (2.72)$$

kterou využijeme k počátečnímu nastavení parametrů následujícím způsobem.

$$\bar{b}^j(0) = d(j) \quad (2.73)$$

$$\bar{x}_i^j(0) = x_i(j) \quad (2.74)$$

$$\sigma_i^j(0) = \frac{[\max(x_i^l : l = 1, 2, \dots, M)] - \min(x_i^l : l = 1, 2, \dots, M)}{M} \quad (2.75)$$

Pro správnou funkci algoritmu je důležitá i volba velikosti kroku μ , která se obvykle provádí experimentálně.

3 Přehled metod detekce novosti

Tato kapitola je věnována přehledu různých vybraných přístupů k detekci novosti, které se v posledních letech používají. Více pozornosti je věnováno algoritmům Learning Entropy (viz kap.3.3.2) a Error and Learning Based Novelty detection (viz kap. 3.3.3) které jsou v kapitole 6 použity pro porovnání úspěšnosti detekce novosti.

Vzhledem k dlouhé historii oblasti detekce novosti vznikla celá řada přehledových publikací, které metody detekce novosti rozdělují do různých skupin. Miljkovič v [12] rozděljuje metody detekce novosti na metody založené na klasifikaci (pravidlové systémy, neuronové sítě a SVM), nejbližší sousedy (vzdálenost, hustota), klastrovací metody, statistické (parametrické a neparametrické) a ostatní (teorie informace, spektrální dekompozice, vizualizace).

Zimek a Filzmozer [13] rozdělují metody detekce novosti opět na celou řadu kategorií. Na metody s učitelem a bez učitele, na metody založené na skóre nebo příslušnosti ke množině, přičemž metody založené na skóre přiřazují datům nějaké číselné ohodnocení, podle kterého se rozhoduje, zda jde o data nové či nikoliv. Metody založené na příslušnosti zařazují data rovnou do jedné ze dvou množin označené „normální“ a „abnormální“. Další rozdělení je na metody parametrické a neparametrické. Mezi parametrické metody autoři řadí metody založené na konkrétních pravděpodobnostních rozložení (např. gaussové). Neparametrické metody jsou metody, které nepracují s parametry pravděpodobnostních rozdělení.

Dvojdílnou, velice obsáhlou přehledovou publikaci napsali Markou a Singh [14, 15]. První díl jejich přehledu je věnován statistickým metodám detekce novosti a druhý díl je věnován metodám založených na neuronových sítích. V první části zabývající se statistickými metodami autoři přicházejí s dělením na metody parametrické (pravděpodobností, Gaussové smíšené modely, skryté markovské modely a testování hypotéz), a neparametrické (kNN, Parzenovo okno, porovnávání řetězců a shlukovací metody). Metody založené na neuronových sítích pak dělí na přístupy využívající vícevrstvý perceptron, metodu podpůrných vektorů, teorii adaptivní rezonance, sítě typu RBF (radial basis function), autoasociátory, Hopfieldovy sítě, oscilující sítě, samoorganizující se mapy, neuronové stromy, sítě s neasociativním učením a ostatní přístupy.

Nejnovější extenzivní publikací je velice rozsáhlá přehledová publikace čítající přes 300 referencí byla vytvořena Pimentelem, D. a L. Cliftonovými a Tarassenkem [16]. Autoři zde dělí

metody detekce novosti na pravděpodobnostní (parametrické a neparametrické), metody založené na vzdálenosti (nejbližší soused, klastrovací metody), metody založené na rekonstrukci (neuronové sítě, metody založené na podprostorech), metody vycházející z teorie informace a doménové metody (SVM).

Následující dvě podkapitoly 3.1 a 3.2 pak vycházejí z dělení, které bylo zavedeno v [14, 15], přičemž důraz je kladen na přístupy, které byly publikovány v několika posledních letech.

3.1 Vybrané statistické metody

V této kapitole jsou podrobněji zmíněny aktuální klasické statistické metody detekce novosti, jmenovitě metody založené na teorii extrémních hodnot, metody využívající gaussovské smíšené modely a skryté Markovské modely. Je zajímavé, že statistické metody, které byli dříve v oblasti detekce novosti dominantní, jsou, co se týče počtu publikací, přechýšleny v posledních 4 letech metodami využívající strojové učení a neuronové sítě.

3.1.1 Metody využívající Gaussovy smíšené modely

Gaussovské smíšené modely (GMM - Gaussian Mixture Models) jsou modely, které se používají pro aproximaci rozdělení pomocí lineární kombinace normálních rozdělení [17]. Podle [18] lze téměř jakoukoliv spojitou funkci hustoty pravděpodobnosti aproximovat s libovolnou přesností pomocí dostatečného počtu gaussovských komponent s vhodně zvolenými středními hodnotami, kovariencemi a váhovacími koeficienty (více např. v [19, 20]). Tyto modely jsou vhodné i pro vícerozměrná data, je-li k dispozici dostatečně velké množství dat pro odhad jejich parametrů. Rozložení pravděpodobnosti klasického GMM je definováno jako

$$f(\mathbf{x}) = \sum_{i=1}^M K_i \mathcal{N}_i(\mathbf{x}) \quad (3.1)$$

přičemž určité úskalí skýtá potřeba apriorní informace o vektoru středních hodnot, kovarianční matici a příslušném váhovacím koeficientu K_i dané i -té gaussovské komponenty.

V oblasti detekce novosti se pak obvykle určí hodnota hustoty pravděpodobnosti GMM daného vzorku dat a podle definovaného prahu se určí zda se jedná o data nová či nikoliv. V [18] používají autoři GMM k monitoringu převodovky. Zpracovávaným signálem jsou naměřené vibrace převodovky. Tento signál je segmentován a na základě segmentů signálů bez poruchy jsou odhadnuty parametry GMM. Při monitoringu je potom pro každý segment spočítána hodnota negativní logaritmicke věrohodnostní funkce (NLL) [21]. Autoři ukazují, že vysoké hodnoty NLL korespondují s poškozením konkrétních zubů převodovky. V [22] používají autoři VB-GMM (Variational Bayesian Gaussian Mixture Model) k detekci poruch plynových turbín. Oproti klasickému GMM, kde je počet gaussovských komponent zvolen apriori, využívá

VB-GMM variační Bayesovské metody k určení optimálního množství komponent [23]. Autoři ukazují, že pomocí zvolené metody jsou s vysokou mírou spolehlivosti rozpoznat různé závady, které korespondují s novostí v měřených datech. Uplatnění GMM je i v oblasti finančnictví, konkrétně v detekci insider tradingu. V příspěvku [24] používají autoři standartní GMM pro modelování poptávky na Tokijské burze a na odhalených případech zneužití důvěrných interních informací, při obchodování s cennými papíry, demonstrují výhody publikovaného přístupu. Úskalím používání složitých GMM je velký počet jejich parametrů. Čím více parametrů zvolený model má, tím více dat je potřeba k jejich odhadu a ověření daného modelu. Pro nalezení parametrů GMM je často používána EM metoda (Expectation-maximization) [25], případně její výpočetně méně náročná modifikace [26]. Další aplikace GMM a smíšených metod lze nalézt v aktuální přehledové publikaci [27].

3.1.2 Metody založené na teorii extrémních hodnot

Teorie extrémních hodnot (EVT) se zabývá distribucí dat s abnormálně velkými nebo malými hodnotami z ocasů generujících rozdělení [28]. Z hlediska teorie extrémních hodnot jsou klíčové dvě věty. Fisher-Tippett-Gnedenkova věta, podle které rozdělení maximum vzorků nezávislých náhodných proměnných se stejným rozdělením po vhodné normalizaci konverguje k jednomu ze tří rozdělení extrémních hodnot (Fréchetovo, Gumbelovo nebo Weibullovo) [29]. Druhou větou je Pickand-Balkema-de Haanova věta [30, 31], která je uvedena v kapitole 4, a která ukazuje důležitost zobecněného Paretova rozdělení pro modelování ocasů různých typů rozdělení.

V [28] autoři porovnávají výsledky na datech pacientů s tremorem získané pomocí EVT a GMM a ukazují, že při použití EVT získají méně falešně pozitivních výsledků. Pro rozlišení zda má pacient tremor používají hodnotu prahu $P = 0.95$. V [32] autoři přicházejí se zobecněním generalizovaného rozdělení extrémních hodnot pro práci s vícerozměrným a vícemodálním rozdělením. Dále navrhuje nové číselné ohodnocení míry novosti, které lze interpretovat jako pravděpodobnost, že extrémní hodnota bude blíž středu rozdělení (ve smyslu Mahalanobisovy vzdálenosti [33]). Použitelnost nového přístupu pak demonstrují na datech pacientů u nichž je nepřetržitě monitorován srdeční tep a rychlost dýchání. Při zachování úspěšnosti detekce skutečně pozitivních pacientů (ve smyslu krizové situace) snižují o 58% falešně pozitivní detekce. V další, obdobné, publikaci [34] pak autoři přicházejí s použitím zobecněného Paretova rozdělení ve vícerozměrných problémech. EVT nalézá své uplatnění i v kombinaci s Kalmanovým filtrem [35]. V [36] autoři detekují novost v datech pomocí AR modelu Kalmanova filtru. Pomocí EVT modelují pravděpodobnost hodnoty koeficientů AR modelu filtru, přičemž pro rozlišení novosti používají velikost prahu $P = 0.95$. Použitelnost uvedeného přístupu demonstrují na umělých datech, ale např. i na datovém souboru obsahující data indexu Dow Jones v období 1972-1975 (pro zajímavost uveďme, že se autorům podařilo detekovat odsouzení spolupracovníků prezidenta Nixona v aféře Watergate, uvalení embarga OPEC na státy podporující

Izrael během Jomkipurské války a rezignaci prezidenta Nixona). V [37] autoři navrhuji novou metodu adaptivní volby prahu pro zobecněné Paretovo rozdělení a ukazují možnost využití v oblasti kybernetické bezpečnosti, kde úspěšně detekují pokus o penetraci sítě. Navrhovaná metoda využívá počtu naměřených dat, požadovaného hodnoty pravděpodobnosti, počtu překročení stávajícího prahu a k němu příslušných hodnot parametrů GPD. Publikace [38] je věnována odhadu parametrů GPD pomocí metody maximální věrohodnosti kombinované s optimalizací hejnem částic (PSO - Particle Swarm Optimization [39]). Autoři ukazují, že jimi navrhovaný přístup přináší výrazné zlepšení v přesnosti odhadu parametrů. Problematicke detekce novosti v point pattern datech (data která jsou tvořena množinami vektorů) se věnují autoři v [40]. Problémem této oblasti je obrovský počet dimenzí a chybějící data v oblastech extrémů, z čehož plyne nutnost pracovat s řídkými daty. Proto autoři navrhuji algoritmus pro extrapolaci modelu abnormálních dat z modelu normálních dat a zakomponování dostupných informací o extrémech z dostupných dat.

3.1.3 Markovovy skryté modely

Skryté Markovovy modely (HMM) se objevují v celé řadě oblastí, jmenujme alespoň, mimo oblasti detekce novosti [14, 16], zpracování řeči [41], rozpoznávání [42], bioinformatiku [43], analýzu časových řad [44] a finančnictví [45]. Skrytým Markovovým modelem formálně rozumíme pěti

$$\vartheta = (S, V, \pi, \mathbf{A}, \mathbf{B}) \quad (3.2)$$

kde S je konečná neprázdná množina stavů, V je neprázdná konečná množina výstupních symbolů, π je vektor počátečních pravděpodobností jednotlivých stavů, \mathbf{A} je matice pravděpodobnosti přechodů mezi jednotlivými stavy a \mathbf{B} je matice pravděpodobností výstupních symbolů pro jednotlivé stavy.

Jednou z možností jak využít HMM pro detekci novosti je přístup, který byl použit pro detekci zneužití telefonních hovorů v [46]. Autoři pro sekvenci stavů určí počet nepravděpodobných přechodů a tento počet použijí jako skóre pro míru novosti v dané sekvenci. Jiného přístupu pro detekci novosti používají i autoři v [47]. Detekují zde potenciální poruchy nebo narušení vodovodního systému. Autoři navrhuji, mimo jiné, využití několika referenčních HMM, kde každý odpovídá určitému časovému intervalu a venkovní teplotě (data vykazují značnou sezónnost). Tím získají několik referenčních modelů, přičemž některé referenční modely obsahují také různé typy anomálií. Při detekci potom sledují počet zda nějaká sekvence stavů odpovídá již známé anomálii, případně určí vzdálenost od testované sekvence ke všem referenčním HMM a pokud vzdálenost překročí předem definovanou vzdálenost, je testovaná sekvence označena jako anomální. V [48] autoři sledují stav ložiska. Používají adaptivní HMM, do něhož přidávají další skryté stavy, které reflektují postupnou degradaci ložiska. Celkový stav ložiska pak určují podle tzv. Health Index, který vyjadřuje vzdálenost hustot pravděpo-

dobností HMM pro současná a historická data. Autoři uvádějí, že jimi navrhovaný přístup dosahuje oproti stávajícím metodám podstatně lepších výsledků. Detekci novosti v datech vibrací převodovky se věnují autoři v publikaci [49]. Z dat vibrací vytvoří spektrogram a určí průběh rychlosti otáčení. Na takto získaná data aplikují PCA [50] a vytvoří odpovídající HMM, který poté využívají k získání diskrepančního signálu. Rozdíl mezi horní a dolní hranicí tohoto signálu a jeho střední hodnotou ve zvoleném okně pak využívají k detekci novosti, která odpovídá vzniklé závadě. Detekci anomálií ve vícerozměrných časových řadách je věnována publikace [51]. Autoři testují různé transformace vícerozměrných řad na jednorozměrné a následnou detekci pomocí HMM. I přes slibné výsledky, upozorňují na zásadní problém, který se týká výpočetní náročnosti jimi navrhovaných přístupů. Zajímavou aplikací HMM je detekce neobvyklého chování námořních lodí [52]. Autoři z naměřených rychlostí, kurzů a poloh vytvoří 5 HMM, kdy jeden odpovídá normálnímu chování a zbylé 4 různým typům neobvyklého chování (příliš vysoká rychlost, neočekávané zastavení, pohyb po spirálové trajektorii, drift lodí). Detekce anomálního chování pak porovná aktuální sekvenci pohybu lodí se všemi HMM a pro každý HMM určí jeho pravděpodobnost. Pokud je pravděpodobnost jednoho z abnormálních modelů větší než pravděpodobnost HMM reprezentujícího normální chování, je chování lodě klasifikováno jako abnormální. Uvedeným přístupem jsou autoři schopni detekovat až 96% lodí a zároveň mají velice nízký počet falešně pozitivních i falešně negativních klasifikací oproti stávajícím metodám. Mezi další zajímavé aplikace HMM v oblasti detekce novosti patří detekce abnormálního chování software [53], detekce SQL injection [54] (pokus o napadení databáze podstrčením nežádoucího kódu přes neošetřený vstup) nebo detekce zneužití předplacených platebních karet [55].

3.2 Vybrané metody využívající strojové učení a neuronové sítě

V této podkapitole je uveden menší souhrn přístupů využívajících různé typy neuronových sítí, které se s rozvojem hlubokého učení stávají stále více populární a to díky výborným výsledkům, kterých, nejen, v oblasti detekce novosti dosahují. Jmenovitě se jedná o metodu podpůrných vektorů, samoorganizující se mapy, konvoluční neuronové sítě a autoenkodéry.

3.2.1 Metoda podpůrných vektorů

Metoda podpůrných vektorů [56] (SVM - support vector machine) bývá některými autory řazena mezi neuronové sítě (např. [15]). Vzhledem k tomu, že lineární verze SVM [57] je podobně jako perceptron [58] lineární klasifikátor a spadá také do oblasti strojového učení [59, 60], zařadili jsme ji do kapitoly věnované neuronovým sítím.

Obecně SVM funguje tak, že rozdělí trénovací data v prostoru na dvě oblasti. Data v každé oblasti náleží do příslušné třídy a jsou oddělena rozhodovací hranicí. Rozhodovací hranice odděluje data tak, že vzdálenost dat z obou různých tříd je od této hranice maximální [61].

Lineární SVM rozdělí data pomocí nadroviny, pokud jsou data lineárně separabilní. Problém nastává v okamžiku, kdy data nejsou plně lineárně separabilní, což může být např. případě dat obsahujících šum.

Nelineární SVM [57] zobrazí data do prostoru o vyšší dimenzi pomocí vhodného nelineárního zobrazení a následně je nalezena nadrovina, která data rozdělí. Zobrazením dat do prostoru o vyšší dimenzi se tedy převede problém na lineárně separabilní. Zobrazením dat do prostoru o vyšší dimenzi se realizuje jádrovou transformací, přičemž jako jádrové funkce se obvykle používá funkce polynomiální, RBF, sigmoida, dvouvrstvý perceptron a podobné. Existují ale i jádrové funkce pracující s řetězci. Podrobnosti o požadavcích pro jádrové funkce jsou uvedeny např. v [62]. Pro snížení výpočetního času, který s přechodem do vyšších dimenzí roste, se využívá tzv. jádrového triku (kernel trick) [63]. I to jsou důvody, proč je SVM velice efektivní a populární metodou v oblasti detekce novosti.

Z pohledu oblasti detekce novosti je klíčová publikace [64], kde autoři využívají SVM následujícím způsobem. Pro trénovací data se nalezne rozhodovací křivka taková, že předem stanovená část trénovacích dat se nalézá v oblasti vymezené rozhodovací hranicí. Detekce novosti je pak realizována tak, veškerá data, která leží vně jsou klasifikována jako nová. Ve studii věnované online detekce poruch chladících jednotek [65] autoři používají rozšířený Kalmanův filtr pro odhad stavu jednotky v kombinaci s rekursivní jednotřídní SVM. Na velkém vzorku dat bez poruchy natrénují SVM, přičemž data pro trénování jsou odhadnuté parametry modelu jednotky. Při monitorování potom v konstantních časových periodách rozšiřují trénovací množinu a přetrénují SVM (princip rekursivní SVM), ovšem pouze v případě, že se v datech ještě neprojevila porucha. Porucha chladící jednotky se projeví změnou parametrů modelu rozšířeného Kalmanova filtru, což je detekováno pomocí SVM. Použitím rozšířeného Kalmanova filtru redukuje oproti stávajícím metodám počet potřebných atributů k vyhodnocování a snižují počet potřebných senzorů k monitoringu. Uvedeným postupem dosahují také vyšší úspěšnost detekce poruchy oproti stávajícím metodám. Autoři uvádějí, že jejich úspěšnost detekce je až 95%. V [66] rozšiřují klasický jednotřídní SVM přístup k detekci novosti o nelineární atributy. Pro naměřená data určí autoři jejich maximum, minimum, střední hodnotu, hodnotu špička - špička, medián, směrodatnou odchylku, kvadratický průměr, šikmost, špičatost a činitel výkonu. K těmto charakteristikám naměřených dat autoři přidávají další nelineární atributy, konkrétně approximate entropy [67], sample entropy [68] a permutační entropii [69]. Navrhovaným přístupem autoři dosahují lepších výsledků při detekci poruchy ložiska oproti stávajícím používaným metodám detekce. V práci [70] používají autoři SVM k detekci nových objektů na snímcích infračerveného kosmického teleskopu (WISE). Autoři testují různé typy jádrových funkcí (lineární, sigmoidní, rbf a polynomiální) na velkém datasetu čítajícím 747 milionů oklasifikovaných objektů a ukazují, že pro hledání nových objektů je nejvhodnější RBF jádrová funkce. Dále uvádějí, že jimi dosažené výsledky jsou lepší, co se týče počtu detekcí neznámých objektů, oproti předchozím publikovaným metodám.

3.2.2 Samoorganizační mapy

Samoorganizační mapy (SOM) [71] jsou umělé neuronové sítě, které více rozměrná vstupní data reprezentují pomocí 1-D nebo 2-D mapy, v podstatě provádějí vektorovou kvantizaci vstupního prostoru. Mapa je tvořena neurony, jejichž vzájemná pozice by měla odpovídat topologii vstupních dat v tom smyslu, že vstupní vektory, které byly podobné (ve smyslu jejich vzdálenosti) by měli být zobrazeny na neurony, které spolu v mapě sousedí [72]. Tyto sítě se učí pomocí kompetitivního učení (bez učitele) a v praxi se často používají pro redukci dimenze zpracovávaných dat [73]. SOM ale nalézají i uplatnění v detekci novosti, o čemž se lze přesvědčit např. v extenzivní přehledové publikaci [72].

V publikaci [74] autoři vytvoří SOM z vícerozměrných dat a na získanou mapu následně použijí shlukovací algoritmus (více o shlukovacích algoritmech např. v [75]). Získané shluky rozdělí do dvou množin, podle velikosti prahu, který odpovídá počtu neuronů v příslušném shluku. Tyto dvě množiny pak reprezentují „velkou část populace“ a „malou část populace“. Následně zavádějí nelineární metriku pro vyhodnocování data, která bere v potaz množství neuronů příslušných danému shluku, vzdálenost vyhodnocovaných dat od daného shluku a příslušnost shluku ke množině „velká část populace“, resp. „malá část populace“. Výsledky navržené metody na několika datasetech porovnávají s dalšími metodami detekce novosti a ukazují, že nově navrženým přístupem dosahují lepších výsledků. V případové studii [76] využívají autoři SOM k detekci podvodů. Pro případ zneužití platebních karet využívají množinu dat, která reprezentuje chování jednoho uživatele. Z těchto dat pak vytvoří SOM a určí centroid reprezentující celou mapu. Podvodné chování je potom detekováno pomocí vzdálenosti od středu centroidu, přičemž autoři vycházejí z předpokladu, že při zneužití karty bude chování vykazovat anomálii oproti chování majitele a z pohledu SOM se bude bod, reprezentující podvodné chování, nalézat v neobvykle velké vzdálenosti od středu centroidu, přičemž neobvykle velká vzdálenost je definována předem zvolenou prahovou hodnotou. V publikaci [77] využívají pro detekci novosti autoři kombinaci SOM a DNF neuronové sítě (dynamic neural field) [78]. SOM slouží k identifikaci atributů pozadí a lze ji použít k rekonstrukci původního pozadí, tak, že rozdíl mezi rekonstruovaným obrazem a původním bude minimální. Při detekci se pak využívá rozdílu mezi rekonstruovaným a původním obrazem. Z tohoto rozdílu je vytvořena saliency mapa [79]. Protože SOM není schopna zrekonstruovat nové objekty v obraze, vznikne v saliency mapě v místech nových objektů oblast pixelů s vysokou hodnotou intenzity. Takto vzniklá mapa je pak zpracována DNF, čímž se odstaní šum kolem hran nového objektu a zvýší se kontrast. SOM organizující mapy se v oblasti detekce novosti používají i pro detekci anomálií v datových tocích z kamerových bezpečnostních systémů [80], pro identifikaci degradace bipolárních tranzistorů [81], k detekci DDoS útoků [82], v analýze zdravotní péče [83] ale i k identifikaci chybných dat v dlouhých datových řadách u kterých docházelo postupem času ke změně metodiky pro získávání těchto dat [84].

3.2.3 Konvoluční neuronové sítě

S rozvojem hlubokého učení se dostali v poslední době do popředí i konvoluční neuronové sítě (CNN). Zásadní publikací z pohledu teorie CNN je potom [85]. Nutno poznamenat, že CNN nenašli své uplatnění jen ve zpracování obrazu, ale také ve zpracování časových řad. V následujícím textu považujeme za CNN jakoukoli neuronovou síť, která má alespoň vrstvu používající konvoluci (což je ve shodě s [86]). Předností CNN je jejich vysoká úspěšnost detekce, respektive klasifikace. Ta je všem dosažena pomocí velkého množství parametrů sítě, z čehož plyne i požadavek na dostatečně velký vzorek dat pro natrénování. Samostatným problémem je pak volba architektury CNN ve smyslu počtu vrstev, velikosti konvolučních jader, volbě pooling vrstvy, učícím algoritmu apod.

S velice zajímavou aplikací CNN pro detekci novosti přicházejí autoři v [87]. Pro monitoring technického stavu staveb využívají data naměřená akcelerometrem. Naměřená data segmentují, a tyto segmenty převedou v časové i frekvenční oblasti převedou na obrazy, které zobrazují průběh amplitudy v čase, respektive výkon pro různé frekvence. Tato reprezentace dat je obvyklá posouzení expertem. Výše uvedeným způsobem předzpracovaná data pak slouží k natrénování CNN. Pro detekci anomálií, které souvisí s potenciální závadou na stavbě pak používají stejným způsobem předpřipravená data, která jsou kontinuálně měřena. Konkrétně jde o data z několika akcelerometrů, které sledují stav mostu. Autoři dále uvádějí, že uvedeným přístupem lze zpracovávat libovolné časové řady. V publikaci [88] autoři používají CNN k detekci defektů na snímcích povrchů. Pro natrénování CNN používají data které neobsahují žádné defekty. Vstupem do neuronové sítě jsou tři obrazy. Prvním a druhým obrazem jsou různé snímky stejného povrchu bez defektu a třetím je obraz jiného povrchu. Trénování sítě pak minimalizuje vzdálenost mezi prvními dvěma obrazy (vstup a prototyp) a maximalizuje vzdálenost mezi prvním a třetím. Při testování je výstupem sítě obraz, jehož intenzita znázorňuje vzdálenost oblasti obrazu od prototypu. Oblasti s vysokou vzdáleností jsou pak oblasti s defektem. Ve studii zabývající se detekcí patologie srdce s využitím fonokardiogramu [89] autoři používají CNN k detekci abnormálních zvuků srdce. Pro natrénování CNN fonokardiogram autoři nasegmentují a pro každý segment vytvoří mel keprstrum, obálku a mel spektrogram. Výsledkem je klasifikátor s přesností 0.815, sensitivitou 0.845 a specifitou 0.785 což jsou výsledky, kterých dosahují nejlepší automatizované klasifikační systémy. Kombinaci autoenkodéru a CNN používají autoři pro detekci nových objektů v obrazech v publikaci [90]. CNN slouží pouze pro rozlišení mezi známou a neznámou třídou. Autoenkodér slouží pro posílení klasifikační schopnosti CNN v tom smyslu, že během předzpracování obrazu pro CNN odstraní šum z obrazů známých tříd a zkreslí obrazy neznámých tříd. Tím je dosažena zvýšená přesnost detekce neznámé třídy pomocí CNN. V [91] používají autoři CNN k detekci abnormálního chování z pohledu denních aktivit. Další oblast využití je detekce novosti předložených dokumentů [92], přičemž využití je např. v systémech kontroly plagiátorství. V

[93] autoři přidávají do latentní oblasti prostoru atributů (což je defakto výstup konvoluční vrstvy) gaussovský šum a tím zvyšují úspěšnost detekce novosti. V [94] pak autoři navrhuji metodiku pro zjištění, která část CNN je zásadní pro rozlišení neznámé třídy.

3.2.4 Autoenkodéry

Autoenkodéry jsou speciálním typem neuronových sítí. Jejich specifikem je, že vstupní i výstupní vrstva obsahují stejný počet neuronů. Počty neuronů ve skrytých vrstvách jsou však menší. Při učení těchto sítí je cílem získat na výstupu enkodéru identický vektor dat jako je na vstupu. Každý autoenkodér můžeme reprezentovat pomocí dvou sítí. Kódovací síť realizuje zobrazení z prostoru vzorů do prostoru příznaků. Vzhledem k rozdílné dimenzi těchto prostorů je možné uvedenou operaci označit jako kódování. Uvažujme vstupní vektor $x \in R^n$. Reprezentace tohoto vektoru ve skryté vrstvě $z(x) \in R^m$ je obecně určená funkcí

$$z(x) = f_1(W_1x + b_1) \quad (3.3)$$

kde f_1 je nelineární aktivační funkce, $W_1 \in R^{n \times m}$ je matice vah a $b_1 \in R^m$ je vektor příslušných biasů. Tato zakódovaná reprezentace vzoru x je zobrazena do prostoru rekonstruovaných vzorů $\hat{x} \in R^n$, přičemž toto zobrazení je realizováno výstupní vrstvou autoenkodéru, tedy

$$\hat{x} = f_2(W_2z(x) + b_2) \quad (3.4)$$

kde f_2 je vhodná funkce, $W_2 \in R^{m \times n}$ je matice vah a b_2 je vektor příslušných biasů [95]. Při učení autoenkodérů se obvykle minimalizuje tzv. chyba rekonstrukce, definovaná jako

$$\mathcal{L}(x, \hat{x}) = \|x - \hat{x}\|^2 \quad (3.5)$$

pomocí metody zpětného šíření (backpropagation).

V [96] používají autoři autoenkodér pro detekci novosti tak, že nejprve autoenkodér naučí a poté mu předkládají různé vzory. Podle chyby rekonstrukce pak usuzují, zda lze předložený vzor klasifikovat jako nový nebo nikoliv. Rozhodovacím kritériem je tedy velikost chyby rekonstrukce \mathcal{L} . Pokud je chyba rekonstrukce větší než zvolená prahová hodnota, pak je vzor klasifikován jako nový. Stejné kritérium pro detekci novosti v audio datech používají i autoři v [97]. Požívají ale modifikovanou strukturu autoenkodéru, jehož skrytá vrstva obsahuje více neuronů než vstupní a výstupní vrstva (denoising autoecoder). Tento typ autoenkodérů bývá využit k odstranění šumu v časových řadách. V [95] navrhuji autoři pro detekci novosti využít odhad hustoty skryté vrstvy. Pokud vstup autoenkodéru má nízkou hustotu ve skryté vrstvě, je tento vstup klasifikován jako nový. Pro modelování hustoty skryté vrstvy autoři navrhuji používat jádrový odhad. Oblastí, ve které nacházejí autoenkodéry uplatnění je i oblast hlubokého učení. V [98] je použita série autoenkodérů, kde vstupem do každého enkodéru je výstup

předcházejícího. Je tedy nutné pracovat s řadou chyb rekonstrukce. Práh pro určení zda-li se jedná o data nová, označme β , pak autoři navrhuji ve tvaru

$$\beta = \alpha \times \text{median}(\mathcal{L}_1, \dots, \mathcal{L}_n) \quad (3.6)$$

kde n je počet použitých autoenkodérů a α je volitelný koeficient. Tento práh je porovnáván s celkovou chybou rekonstrukce sítě autoenkodérů. Ve článku [99] je navržen nový typ autoenkodéru DGP-AE (Deep Gaussian Process autoencoder), jehož kódovací a dekódovací část využívá gaussovských procesů. Předností uvedené sítě je možnost využití v oblastech, kde se pracuje s daty kategorickými i numerickými. Autoenkodéry nalézají uplatnění i v oblastech zpracování obrazu. Pro kategorizaci objektů v obraze a detekci neznámých objektů používají autoři v [100] autoenkodér s dvěma dekódovacími sítěmi, pro jehož učení využívají algoritmus GZSL (Generalized Zero Shot Learning). Pro detekci nového objektu pak navrhuji využívat k nově zavedenou funkci, která využívá chyby rekonstrukce obou dekódovacích vrstev. Řada přístupů k detekci novosti s využitím autoenkodérů využívá různé modifikované struktury, ať již to jsou např. rekurentní autoenkodéry [101, 102], variační autoenkodéry [103] nebo tzv. self-adversarial variační autoenkodéry [104].

3.3 Další vybrané metody využívající adaptivní filtry

Vzhledem k tomu, že tématem předkládané dizertační práce jsou adaptivní systémy, byla tato kapitola vyčleněna metodám detekce novosti, které využívají právě adaptivní filtry. Zvláštní pozornost a podrobný popis se pak týká algoritmům Learning Entropy a Error and Learning Based Novelty Detection, které jsou v kapitole 6 použity k porovnání výsledků.

3.3.1 Metoda set-membership

Metoda set-membership [105, 106] je metoda použitá nejen ke zrychlení adaptace filtru ale i pro detekci novosti. Tato metoda pracuje s chybou predikce a adaptivním filtrem. Základní myšlenka je, že pokud je chyba predikce filtru e větší než nějaká prahová hodnota γ , dojde k adaptaci tohoto filtru. Podmínka pro adaptaci filtru je tedy ve tvaru

$$|e(k)| > \gamma. \quad (3.7)$$

V okamžiku, kdy dochází k adaptaci filtru lze hovořit o novosti v datech. V originálním článku [106] autoři metodu využívají ke zrychlení konvergence NLMS algoritmu a přicházejí s novým způsobem adaptace vah filtru, algoritmem set-membership-NLMS (SM-NLMS).

Diniz a Yazdanpanah v [107] používají metodu set-membership a navrhuji nový algoritmus double-threshold-SM-NLMS, který využívají k ořezávání datasetů. Autoři zde rozdělují data

do dvou množin. V první množině jsou data přinášející nějakou informaci a ve druhé data které novou informaci neunesou. Základní myšlenka je, že pokud jsou pro adaptivní filtr data nová, bude chyba predikce v určitých mezích. Tyto meze jsou dány dvěma prahovými hodnotami a pokud se chyba predikce filtru e nachází v těchto mezích, přinášejí data z pohledu celého datasetu nějakou užitečnou informaci. Naopak data, pro která je chyba predikce adaptivního filtru malá tuto informaci nepřinášejí a stejně tak data, pro které je chyba predikce příliš velká jsou outlier a tedy nepřinášejí žádnou užitečnou informaci. Oproti klasickému SM-NLMS algoritmu, který pracuje s jedním prahem tedy autoři navrhuji prahy dva. Podmínka pro adaptaci filtru je potom ve tvaru

$$\bar{\gamma}_1 \leq |e(k)| \leq \bar{\gamma}_2 \quad (3.8)$$

Výsledná metoda slouží ke snížení výpočetního času pro adaptaci filtru a zároveň data klasifikuje do dvou výše zmíněných množin.

Algoritmy založené na metodě set-membership našly široké uplatnění v řadě oblastí. Jmenujme alespoň oblast mobilní robotiky, kde se používají např. při fúzi dat z vícero senzorů [108], při lokalizaci a řízení podvodních robotů [109] nebo při detekci poruch [110].

3.3.2 Algoritmus Learning Entropy

Algoritmus Learning Entropy (LE) je algoritmem, který využívá adaptivních filtrů (obvykle LNU a HONU) k detekci novosti v datech [111, 112]. Autoři ukazují, že ne vždy, je novost v datech korelována s chybou predikce, a k detekci novosti navrhuji využívat přírůstky vah adaptivních filtrů $\Delta \mathbf{w}$, které jsou získány použitím gradientní metody a adaptací s každými novými daty. V první publikaci [111] je popsán multifraktální přístup, který využívá uspořádané množiny různých hodnot prahů

$$\boldsymbol{\alpha} = \{\alpha_1, \alpha_1, \dots, \alpha_{n_\alpha}\}, \alpha_1 > \alpha_2 > \dots > \alpha_{n_\alpha} \quad (3.9)$$

přičemž $\forall i \in \{1, 2, \dots, n_\alpha\} : \alpha_i \in R$. Míra novosti „Approximate Individual Sample Learning Entropy“ (AISLE) je potom definovaná jako

$$E_A(k) = \frac{1}{n_w \cdot n_\alpha} \sum_{i=1}^{n_w} \sum_j^{n_\alpha} f(|\Delta w_i(k)|, \alpha_j) \quad (3.10)$$

kde funkce f je

$$f(\Delta w_i, \alpha_j) = \begin{cases} 1 & \text{if } |\Delta w_i(k)| > \alpha_j \cdot \overline{|\Delta w_i^M(k)|} \\ 0 & \text{if } |\Delta w_i(k)| \leq \alpha_j \cdot \overline{|\Delta w_i^M(k)|} \end{cases} \quad (3.11)$$

a n_w je počet adaptivních parametrů filtru, n_α je počet citlivostních prahů (respektive počet prvků množiny $\boldsymbol{\alpha}$) a $\overline{|\Delta w_i^M(k)|}$ průměrná hodnota přírůstku i -té adaptivní výhy v plovoucím okně délky M . Člen $1/(n_w \cdot n_\alpha)$ normalizuje hodnotu AISLE takže $E_A \in (0, 1)$. Toto hodnotu

lze interpretovat tak že pokud se blíží 0, váhy se téměř nemění a z pohledu adaptace filtru se příliš neliší od minulých dat. Naopak vysoké hodnoty E_A znamenají, že dochází k velkým změnám velkého množství adaptivních vah a data se nějakým způsobem liší od předchozích.

V publikaci [112] pak autoři navrhuji tzv. přímý algoritmus pro výpočet LE. Ta je vypočítána jako

$$LE(k) = \sum_{i=1}^{n_w} \max\{0, z(|\Delta w_i(k)|) - \beta\}; \quad (3.12)$$

kde funkce z je označovaná jako speciální „z-score“ a je definovaná jako

$$z(|\Delta w_i(k)|) = \frac{|\Delta w_i(k)| - \overline{|\Delta \mathbf{w}_i^M(k-1)|}}{\sigma(|\Delta \mathbf{w}_i^M(k-1)|)} \quad (3.13)$$

kde

$$\overline{|\Delta \mathbf{w}_i^M(k-1)|} = \frac{\sum_{j=1}^M w_i(k-j-m)}{M} \quad (3.14)$$

je průměr hodnot přírůstku adaptivních vah, přičemž parametr M je délka plovoucího okna, z kterého je spočítán průměr přírůstků i -té váhy a parametr m je volitelný parametr pro data které vykazují periodicitu a $\sigma(|\Delta \mathbf{w}_i^M(k-1)|)$ je směrodatná odchylka těchto M přírůstků. Pro takto definovanou LE pak platí, že $E(k) \in \langle 0, +\infty \rangle$. Tento přístup umožňuje detekovat změny adaptivních parametrů, které jsou větší než jejich průměr zvětšený o β násobek jejich směrodatné odchylky a zároveň obchází nutnost výpočtu přes několik citlivostních prahů jako v případě 3.10. Pokud bychom chtěli detekovat i neobvykle malé změny adaptivních parametrů, navrhuji autoři vztah 3.12 pro výpočet LE modifikovat na tvar

$$E(k) = \sum_{i=1}^{n_w} z(|\Delta w_i(k)|); E \in R \quad (3.15)$$

který v případě negativních hodnot $z(|\Delta w_i(k)|)$ nemusí poskytovat jasnou hranici mezi obvyklými a neobvyklými změnami adaptivních parametrů.

3.3.3 Algoritmus Error and Learning Based Novelty Detection

Algoritmus Error and Learning Based Novelty Detection (ELBND) [113, 114] je dalším z algoritmů detekce novosti, které využívají adaptivních filtrů, jejichž parametry jsou adaptovány vždy s novými daty. Každý naměřený vzorek dat je popsán vektorem hodnot, na základě přírůstku adaptivních vah a chyby filtru, definovaným jako

$$elbnd(k) = \Delta \mathbf{w} \cdot e(k) \quad (3.16)$$

který popisuje novost v daném vzorku dat. Skalární hodnotu popisující novost ve vzorku dat pak autoři navrhuji získat buď jako maximum z absolutních hodnot vektoru $elbnd(k)$, takže

skalární míra novosti $ELBND(k)$ je definovaná jako

$$ELBND(k) = \max_{1 \leq i \leq n_w} |\Delta w_i(k) \cdot e(k)|. \quad (3.17)$$

Další možností je určit míru novosti jako součet absolutních hodnot jednotlivých přírůstků násobených chybou filtru, tedy

$$ELBND(k) = \sum_{i=1}^{n_w} |\Delta w_i(k) \cdot e(k)| \quad (3.18)$$

Uvedený algoritmus, využívající vztah 3.17, byl úspěšně použit k adaptivní klasifikaci EEG pacientů s demencí [115].

4 Zobecněné Paretovo rozdělení

Přestože je normální rozdělení historicky často používané rozdělení v oblasti detekce novosti (zde bychom měli zmínit Grubbsův test [116] který se stal základním kamenem celé řady metod detekce novosti, resp. odlehlých vzorků), nemusí být vždy vhodné. Někdy může být střední hodnota a symetrický rozptyl zavádějící, ve smyslu výpovědní hodnoty o rozdělení sledovaných dat [117]. Zvláště, když je předmětem zájmu právě ocas rozdělení, z něhož data pocházejí.

Uveďme zde Pickland - Balkema - de Haanovu větu [30, 31] (někdy označována jako druhá věta teorie extrémních hodnot) podle které, pro řadu X_1, X_2, \dots nezávislých náhodných veličin se stejným rozdělením F pro které je F_u rozdělení jejich excesů přes mez u (tedy hodnot které překračují u) platí

$$F_u(x) \rightarrow GPD(\xi, \mu, \sigma)(x), \text{ pro } u \rightarrow \infty \quad (4.1)$$

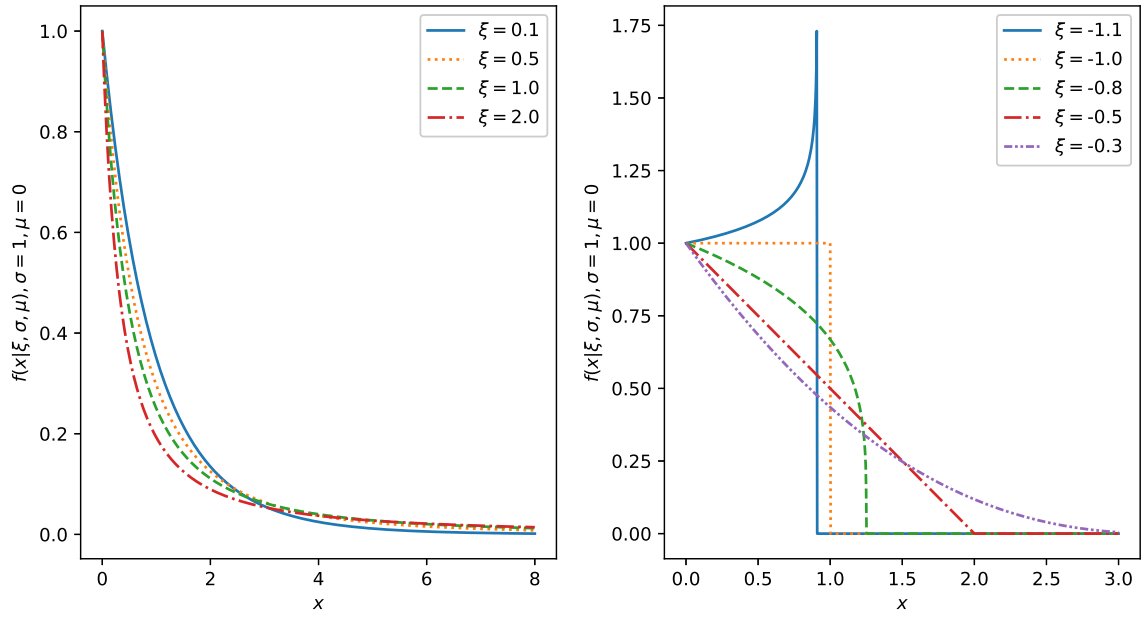
kde GPD je zobecněné Paretovo rozdělení a F_u je distribuční funkce hodnot překračujících práh u definovaná jako

$$F_u(x) = P(X - u \leq x, X > u) = \frac{F(u+x) - F(u)}{1 - F(u)} \quad (4.2)$$

pro $0 \leq x \leq x_F - u$, kde x_F je pravý koncový bod distribuční funkce F . Funkce hustoty pravděpodobnosti GPD je potom definovaná jako

$$f_{(\xi, \mu, \sigma)}(x) = \begin{cases} \frac{1}{\sigma} \left(1 + \frac{\xi(x-\mu)}{\sigma} \right)^{\left(-\frac{1}{\xi}-1\right)} & \text{for } \xi \neq 0, \\ \exp\left(-\frac{x-\mu}{\sigma}\right) & \text{for } \xi = 0. \end{cases} \quad (4.3)$$

kde obecně $\mu \in (-\infty, +\infty)$ je parametrem lokace, $\sigma \in (0, \infty)$ je parametrem měřítka a $\xi \in (-\infty, \infty)$ je parametr tvaru. Odpovídající distribuční funkce je potom ve tvaru



Obrázek 4.1: Hustoty pravděpodobnosti zobecněného Paretova rozdělení pro různé hodnoty parametru ξ a zvolené hodnoty parametrů $\sigma = 1$, $\mu = 0$

$$F_{(\xi, \mu, \sigma)}(x) = \begin{cases} 1 - \left(1 + \frac{\xi(x-\mu)}{\sigma}\right)^{-\frac{1}{\xi}} & \text{for } \xi \neq 0, \\ 1 - \exp\left(-\frac{x-\mu}{\sigma}\right) & \text{for } \xi = 0. \end{cases} \quad (4.4)$$

Důležitým faktem je, že nosič zobecněného Paretova rozdělení je omezený a to tak, že $x \geq \mu$ pro $\xi \geq 0$, a $\mu \leq x \leq \mu - \sigma/\xi$ pro $\xi < 0$ přičemž $\mu \in R$, $\sigma > 0$, a $\xi \in R$.

Obrázek 4.1 zobrazuje hustoty pravděpodobnosti GPD s různými parametry ξ , které mají demonstrovat možnosti použití GPD pro modelování různých typů ocasů. Pro $\xi = 1$ je GPD ekvivalentní s rovnoměrným rozdělením, pro hodnotu $\xi = 0$ je exponenciálním rozdělením, pro $\xi = -0.5$ je trojúhelníkovým rozdělením, pro $-0.5 < \xi < 0$ je rozdělením s lehkým ocasem (např. normální rozdělení, Gumbelovo rozdělení), pro $\xi > 0$ je rozdělením s těžkým ocasem (např. Paretovo rozdělení, logaritnicko-normální rozdělení, Studentovo rozdělení) a pro $\xi < -1$ je rozdělením s kompaktním nosičem (např. beta rozdělení). Vzhledem k tomu, že v aplikacích s reálnými daty se vyskytuje celá řada rozdělení, je GPD, díky své výše demonstrované univerzalitě, vhodným rozdělením pro modelování ocasů jiných rozdělení [118–120].

4.1 Metoda Peak-over-Threshold

Zásadním problémem při hledání parametrů GPD je volba vhodné hodnoty prahu z . Metody, které řeší uvedený problém, jsou nazývány Peak-Over-Threshold (POT). V případě, že je hodnota prahu z příliš vysoká, jsou parametry GPD velmi proměnlivé, protože existuje příliš málo dat, které zvolenou prahovou hodnotu překročí. Pokud je hodnota prahu z naopak příliš nízká, není aproximace ocasu generujícího rozdělení spolehlivá. Z tohoto pohledu, je tedy vhodná volba hodnoty prahu z pro kvalitu fitu GPD zásadní. Existuje celá řada metod pro výběr hodnoty prahu (více viz [121]). Některé z metod POT jsou však výpočetně náročné, případně poskytují výsledky, jež musí být vyhodnoceny expertem. Z těchto důvodů nejsou příliš vhodné pro algoritmus detekce novosti, který by vyhodnocoval data v reálném čase. Proto se zdá vhodné využít některé z metod „Rule of Thumb“. Z těchto metod se osvědčily tři různé metody.

Nechť l je počet vzorků, které jsou použity k odhadu parametrů GPD a n_s je celkový počet vzorků, které jsou k dispozici. Potom podle [122, 123, 125] můžeme určit počet vzorků potřebných k odhadu parametrů GPD jako:

$$l_1 = \lceil 0.1 \cdot n_s \rceil \quad (4.5)$$

$$l_2 = \lceil \sqrt{n_s} \rceil \quad (4.6)$$

$$l_3 = \left\lceil \frac{\sqrt[3]{n_s^2}}{\log(\log(n_s))} \right\rceil \quad (4.7)$$

Ještě poznamenejme, že pro odhad parametrů GPD se využívají vzorky s nejvyšší hodnotou a symbolem $\lceil \cdot \rceil$ rozumíme horní celou část daného čísla.

Metoda POT je zásadním prvkem v algoritmu Extreme Seeking Entropy (viz kapitola 5), kde je použita pro rozdělení přírůstků adaptivních vah filtru $|\Delta w_k(k)|$ do příslušných množin H_i respektive L_i .

4.2 Metody odhadu parametrů zobecněného Paretova rozdělení

Odhad parametrů zobecněného Paretova rozdělení není triviální problém, proto existuje celá řada metod, které tento problém řeší. V následující textu je věnována pozornost metodě momentů (MOM) [124], metodě maximální věrohodnosti (ML) [125] a metodě kvazi-maximální věrohodnosti (QML) [126]. Z nověji publikovaných metod uveďme alespoň metodu věrohodných momentů [127], metodu vážených nelineárních čtvců momentů [128, 129]. Přehled dalších metod nabízí např. publikace [130].

4.2.1 Metoda maximální věrohodnosti

Metoda maximální věrohodnosti (maximum likelihood) je jednou ze zásadních statistických metod (pro zajímavost: byla použita i Johannem Carlem Friedrichem Gaussem nebo Pierre-Simon Laplacedem, více viz [131]). Slouží k odhadu parametrů různých rozdělení pravděpodobnosti [132]. Pro odhad parametrů je použita tzv. věrohodnostní funkce definovaná jako

$$\mathcal{L}(\theta|X_1, \dots, X_n) = \prod_{i=1}^n f(X_i|\theta) \quad (4.8)$$

kde $\theta \in \Omega$ je vektor odhadu parametrů, Ω je množina parametrů rozdělení a X_1, \dots, X_n je soubor nezávislých náhodných veličin se stejným rozdělením a neznámou hustotou pravděpodobnosti. Pro maximálně věrohodný odhad $\hat{\theta}$ pak platí

$$\hat{\theta} = \arg \max_{\theta \in \Omega} \mathcal{L}(\theta|X_1, X_2, \dots, X_n). \quad (4.9)$$

Tento vektor parametrů rozdělení tedy maximalizuje věrohodnostní funkci. V praxi se často pracuje s logaritmem věrohodnostní funkce, takže problém je formulován jako

$$\log \mathcal{L}(\theta|X_1, \dots, X_n) = \sum_{i=1}^n \log f(X_i|\theta) \quad (4.10)$$

Věrohodnostní funkce pro zobecněné Paretovo rozdělení byla zformulovaná DuMouchelem [125] ve tvaru

$$\log \mathcal{L}(\sigma, \gamma|x_1, \dots, x_n) = -n \log \sigma + \frac{1-\gamma}{\gamma} \sum_{i=1}^n \log \left(1 - \frac{\gamma}{\sigma} (x_i - \mu) \right) \quad (4.11)$$

přičemž podle [133] je pro $-0.5 < \xi < 0$, při zachování určitých podmínek, maximálně věrohodný odhad asymptoticky normální a asymptoticky vydatný. Problém může pro hodnoty $\xi < -0.5$ kdy maximálně věrohodný odhad nemusí existovat [134]. Optimalizační problém 4.8 pro GPD s věrohodností funkcí ve tvaru 4.11 obvykle řeší nějakou numerickou metodou.

4.2.2 Metoda momentů

Při použití metody momentů (MOM) je k odhadu parametrů GPD využito prvního a druhého momentu [124]. Abychom poukázali na problém, který je s použitím MOM spojen, uvažujme nyní dvou parametrické GPD, pro které je $\mu = 0$, $\xi > 0$ a $x \geq 0$. Pro první moment potom dostaneme přímo podle definice

$$E(X) = \int_0^\infty x \frac{1}{\sigma} \left(1 + \frac{\xi x}{\sigma} \right)^{\left(-1-\frac{1}{\xi}\right)} dx = \sigma^{\frac{1}{\xi}} \int_0^\infty x (\xi x + \sigma)^{-\frac{1}{\xi}-1} dx = \sigma^{\frac{1}{\xi}} \left[-\frac{x}{(\xi x + \sigma)^{\frac{1}{\xi}}} \right]_0^\infty -$$

$$\begin{aligned}
 -\sigma^{\frac{1}{\xi}} \int_0^\infty -\frac{x}{(\xi x + \sigma)^{\frac{1}{\xi}}} dx &= \sigma^{\frac{1}{\xi}} \left[-\frac{x}{(\xi x + \sigma)^{\frac{1}{\xi}}} \right]_0^\infty + \sigma^{\frac{1}{\xi}} \left[\frac{(\xi x + \sigma)^{1-\frac{1}{\xi}}}{\xi(1-\frac{1}{\xi})} \right]_0^\infty = \\
 &= \left[\frac{\sigma^{\frac{1}{\xi}}(x + \sigma)}{(\xi - 1)(\xi x + \sigma)^{\frac{1}{\xi}}} \right]_0^\infty = \frac{\sigma}{1 - \xi}
 \end{aligned} \tag{4.12}$$

přičemž z předposledního výrazu je patrné, že konečnou hodnotu získáme pouze pro hodnoty $0 < \xi < 1$. Obdobně pro rozptyl GPD

$$\begin{aligned}
 Var(X) &= \int_0^\infty x^2 \frac{1}{\sigma} \left(1 + \frac{\xi x}{\sigma} \right)^{(-1-\frac{1}{\xi})} dx - \frac{\sigma^2}{(1 - \xi)^2} = \\
 &= \left[\frac{\sigma^{\frac{1}{\xi}}((\xi - 1)x^2 - 2\sigma x - 2\sigma^2)}{(\xi - 1)(2\xi - 1)(\xi x + \sigma)^{\frac{1}{\xi}}} \right]_0^\infty - \frac{\sigma^2}{(1 - \xi)^2} = \frac{\sigma^2}{(1 - \xi)^2(1 - 2\xi)}
 \end{aligned} \tag{4.13}$$

přičemž z předposledního výrazu je patrné, že konečnou hodnotu získáme pouze pro hodnoty $0 < \xi < 0.5$.

Obecný r -tý moment GPD existuje pokud je splněna podmínka $\xi < 1/r$. První čtyři momenty GPD (průměr, rozptyl, koeficienty šikmosti a špičatosti), jsou definovány jako

$$E(X) = \mu + \frac{\sigma}{1 - \xi} \tag{4.14}$$

$$Var(X) = \frac{\sigma^2}{(1 - 2\xi)(1 - \xi)^2} \tag{4.15}$$

$$Skew(X) = \frac{2(1 + \xi)\sqrt{1 - 2\xi}}{1 - 3\xi} \tag{4.16}$$

$$Kurt(X) = \frac{2(1 - 2\xi)(2\xi^2 + \xi + 3)}{(1 - 3\xi)(1 - 4\xi)} - 3 \tag{4.17}$$

s přihlédnutím k omezení, že $\xi < 1/r$. Řešením soustavy rovnic 4.14 a 4.15 dostaneme odhady parametrů GPD jako

$$\sigma = \frac{1}{2}E(X) \left(\frac{E(X)^2}{Var(X)} + 1 \right) \tag{4.18}$$

$$\xi = -\frac{1}{2} \left(\frac{E(X)^2}{Var(X) - 1} \right) \tag{4.19}$$

kde $E(X)$ je střední hodnota vzorku dat a $Var(X)$ je rozptyl tohoto vzorku dat. Metoda MOM je použitelná pouze pro hodnoty parametru $\xi < 0.5$. Pro $\xi \geq 0.5$ není splněná podmínka pro existenci druhého momentu.

4.2.3 Kvazi-ML metoda

Kvazi-ML metoda odhadu parametrů GPD kombinuje metodu maximální věrohodnosti pro $\xi > -0.75$ a modifikovanou metodu maximální věrohodnosti pro $\xi \leq -0.75$ [126]. Protože neznáme přesnou hodnotu parametru ξ je rozhodnutí, kterou z metod použít provedeno na empirickém základě. Nejprve uvažujme posloupnost hodnot (x_1, \dots, x_n) . Metodu kvazi-ML lze popsat v následujících třech krocích.

1. Výpočet

$$\xi = \frac{-1}{n-1} \sum_{i=1}^{n-1} \ln \left(1 - \frac{x_1}{\max x_1, \dots, x_n} \right) \quad (4.20)$$

a

$$Z = 1 - \frac{\sum_{i=1}^n x_i}{2n\bar{x}^2} \quad (4.21)$$

kde \bar{x} je střední hodnota posloupnosti hodnot (x_1, \dots, x_n) .

2. Pokud je $\xi < 0.75$ a $Z < 0.2$ použij metodu maximální věrohodnosti.
3. Jinak určíme odhad parametru σ jako

$$\sigma = \xi \cdot \max(x_1, \dots, x_n) \quad (4.22)$$

Z hlediska výpočetní náročnosti a možností hodnot, kterých může parametr ξ nabývat, je uvedená hodnota dobrým kompromisem mezi metodou maximální věrohodnosti a metodou momentů.

5 Algoritmus Extreme Seeking Entropy

V této kapitole je představen algoritmus pro detekci novosti nazvaný Extreme Seeking Entropy (ESE), který tvoří hlavní teoretický výsledek předkládané dizertační práce (publikováno v [V1]). Navržený algoritmus, vychází z předpokladu, že novost v datech se projeví neobvykle velkými přírůstky vah adaptivního filtru, který danou řadu dat modeluje.

Nejprve uvažujme myšlenkový experiment, ve kterém máme dokonale nastavený adaptivní filtr, jehož chyba predikce e je nulová pro všechny vstupní hodnoty. Potom přírůstky adaptivních vah tohoto filtru budou nulové. V případě, že dojde k nějaké změně v generátoru dat pro tento filtr, začne se filtr opět adaptovat což vyústí v nenulové změny adaptivních vah, které budou reflektovat novost způsobenou změnou vlastností daného generátoru.

V publikacích [111, 112], kde autoři představují algoritmus Learning Entropy, je zmíněna obecná míra snahy adaptivního filtru o adaptaci, L , která slouží k vyhodnocení neobvykle velkých přírůstků adaptivních vah a je definovaná jako

$$L(k) = A(f(\Delta \mathbf{w})) \quad (5.1)$$

kde A je obecně nějaká agregační funkce a funkce f je funkce která nějakým způsobem kvantifikuje odchylku v adaptaci adaptivních parametrů filtru.

Nejprve uvažujme, že hodnota funkce f , která slouží k vyhodnocení neobvykle velkých přírůstků by měla mít neobvykle velkou (nebo neobvykle malou) hodnotu v okamžiku, kdy přírůstky adaptivních vah filtru $\Delta \mathbf{w}$ jsou neobvykle velké. Dalším požadavkem je, aby tato funkce zohledňovala i nějakou historii těchto přírůstků. Přirozeně se tedy nabízí nějaká vhodná distribuční funkce f_{cdf} . S ohledem na požadavek, že neobvykle velké přírůstky vah by měli být reflektovány neobvykle velkou hodnotou míry snahy o adaptaci, nabízí se jako vhodná funkce A , kterou uvažujeme ve tvaru

$$A(f(\Delta \mathbf{w}|(k))) = -\log \prod_{i=1}^n (1 - f_{cdf_i}(|\Delta w_i(k)|)). \quad (5.2)$$

Uvedená agregační funkce A má pro neobvykle velké přírůstky vah neobvykle velkou kladnou hodnotu, pokud jsou hodnoty f_{cdf} blízké 1. Člen $1 - f_{cdf}$ je vlastně komplementární distribuční funkcí (funkce přežití, spolehlivostní funkce). Výhodou uvedeného přístupu je absence potřeby nastavovat několik prahů pro detekci.

Jak již bylo uvedeno, cílem je tedy vyhodnotit neobvykle velké přírůstky vah adaptivního systému. Nejprve je nutné určit nějaký práh¹ z , podle kterého můžeme přírůstky adaptivních vah filtru rozdělit do dvou množin. Množinu, která bude obsahovat přírůstky menší než je zvolený práh z označíme L . Přírůstek, který je větší nebo roven hodnotě prahu z označíme H . Uvažujme, že obě množiny existují pro každou adaptivní váhu, potom pro i -tou adaptivní váhu zvolíme práh z_i tak, že velikosti přírůstku této váhy náleží do jedné ze dvou množin, tak, že:

$$\forall |\Delta w_i| < z_i \in L_i \quad (5.3)$$

$$\forall |\Delta w_i| \geq z_i \in H_i \quad (5.4)$$

Vzhledem k výše zmíněnému předpokladu o velikosti změn vah adaptivního filtru a novosti v datech, uvažujme, že přírůstky náležející množině L_i pravděpodobně neobsahují informaci o novosti během adaptace, a proto nebudou vyhodnocovány. Množina H_i by měla obsahovat přírůstky adaptivních vah filtru, u kterých lze očekávat, že mohou nést nějakou informaci o novosti v datech.

Nyní můžeme zavést novou míru, kterou nazvěme Extreme Seeking Entropy (ESE), definovanou jako

$$ESE(|\Delta \mathbf{w}(k)|) = -\log \prod_{i=1}^{n_w} (1 - f_{cdf_i}(|\Delta w_i(k)|)) \quad (5.5)$$

kde

$$f_{cdf_i}(|\Delta w_i(k)|) = \begin{cases} 0, & |\Delta w_i(k)| \in L_i \\ F_{(\xi_i, \mu_i, \sigma_i)}(|\Delta w_i(k)|), & |\Delta w_i(k)| \in H_i. \end{cases} \quad (5.6)$$

a funkce $F_{(\xi_i, \mu_i, \sigma_i)}$ je distribuční funkce zobecněného Paretova rozdělení (GPD), představeného v předchozí kapitole 4. Přírůstky adaptivních vah, které jsou menší než hodnota prahu získaného metodou POT, nezmění hodnotu ESE . Přírůstky vah, které mají velice malou pravděpodobnost a spíše nesou nějakou informaci o novosti v datech, způsobí velký nárůst hodnoty ESE . Zásadním aspektem navrhovaného algoritmu je, že mohou být vyhodnocovány buď všechny získané přírůstky vah, anebo lze vyhodnocovat pouze nejnovějších n_s vzorků a na ně aplikovat metodu POT. Novost potom můžeme vnímat v kontextu těchto n_s vzorků.

Navrhovaný algoritmus pro detekci novosti v datech je popsán následujícím pseudokódem.

¹Pozn. Volba hodnoty prahu přímo souvisí s volbou metody POT (viz kapitola 4).

Algoritmus 1 Extreme Seeking Entropy

```

1: nastavení  $n_s$  a výběr metody POT
2: počáteční nastavení parametrů  $\xi_i, \mu_i, \sigma_i$  GPD pro každý adaptivní parametr
3: for vzorek  $y(k)$  do
4:   výpočet změny adaptivních parametrů filtru  $\Delta \mathbf{w}(k)$ 
5:   aplikace metody POT
6:   if  $|\Delta w_i|(k) \in H_i$  then
7:     výpočet parametrů  $\xi_i, \mu_i, \sigma_i$  pro příslušné GPD
8:   end if
9:   výpočet hodnoty  $ESE$  podle rovnice 5.5
10: end for

```

Výhodou navrhovaného algoritmu je, že jediným volitelným parametrem je parametr n_s , který ale v případě potřeby můžeme vynechat a zpracovávat všechny historicky dostupné přírůstky adaptivních parametrů. Dále je nutné zvolit vhodnou metodu POT.

Určitou limitací uvedeného algoritmu je, že pro získání první hodnoty ESE potřebujeme nějakou apriorní informaci o parametrech GPD rozdělení. Konkrétněji, pro každou i -tou adaptivní váhu musíme mít k dispozici odhad parametrů ξ_i, μ_i, σ_i . Vzhledem k tomu, že adaptivní filtr má n_w různých adaptivních vah, potřebujeme získat odhad $3 \cdot n_w$ parametrů aby začal algoritmus ESE poskytovat první hodnoty. Pokud není k dispozici žádná apriorní informace o hodnotách těchto parametrů, potřebujeme získat alespoň n_s vzorků, na jejichž základě můžeme získat první hodnotu ESE . Dalším úskalím je proměnlivost parametrů nebo typu rozdělení pravděpodobnosti přírůstků adaptivních vah v čase.

6 Výsledky detekce novosti algoritmem Extreme Seeking Entropy

V dalším textu jsou shrnuty výsledky algoritmu Extreme Seeking Entropy, kterých bylo v rámci práce na dizertační práci dosaženo. Výsledky použití ESE pro detekci perturbace v chaotické časové řadě Mackey-Glass je uvedena v podkapitole 6.1. Dále jsou uvedeny výsledky pro: detekci změny rozptylu šumu v náhodném datovém toku (podkapitola 6.2), detekci skokové změny parametrů generátoru signálu (podkapitola 6.3), detekci náhle absence šumu (podkapitola 6.4), detekci změny trendu (podkapitola 6.5) a detekci epilepsie v záznamu myšího EEG (viz podkapitola 6.6). Dále jsou uvedeny výsledky vyhodnocení úspěšnosti detekce skokové změny parametrů generátoru signálu (podkapitola 6.7) a úspěšnosti detekce skokové změny trendu (podkapitola 6.8). V posledních dvou kapitolách je potom vyhodnocení ROC křivky (receiver operating characteristic) pro detekci změny trendu (podkapitola 6.9) a experimentální vyhodnocení výpočetní náročnosti metod odhadu parametrů zobecněného Paretova rozdělení (podkapitola 6.10).

Výsledky v podkapitolách 6.1-6.8 byly publikovány v [V1]. Výsledky v podkapitolách 6.9 a 6.10 pak v publikacích [V2],[V3].

6.1 Chaotická časová řada Mackey-Glass a detekce perturbace

Tento experiment byl proveden pro porovnání s výsledky uvedenými v publikaci [111], která je první publikací o algoritmu LE (viz kapitola 3.3.2). Experiment spočívá v detekci perturbovaného vzorku v chaotické časové řadě, která je výsledkem řešení Mackey-Glassovy rovnice [135].

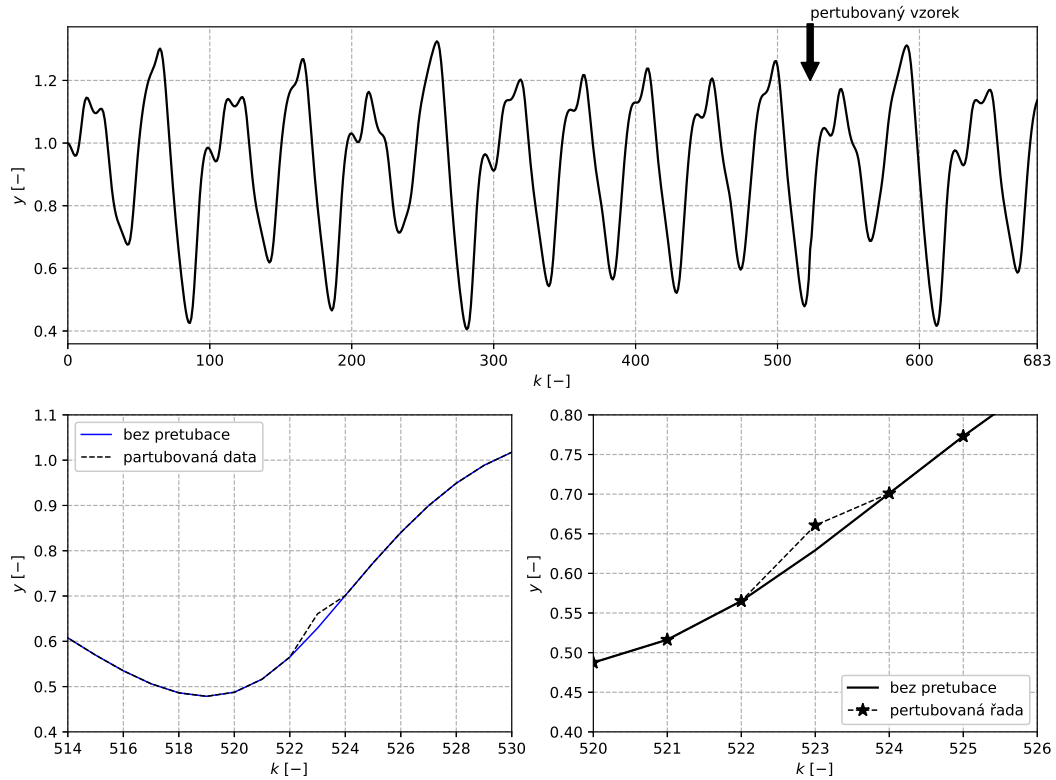
$$\frac{dy(t)}{dt} = \beta \cdot \frac{y(t - \tau)}{1 + y^\alpha(t - \tau)} - \gamma y(t) \quad (6.1)$$

přičemž parametry $\alpha = 10$, $\beta = 0.2$, $\gamma = 0.1$ and $\tau = 17$ byly vybrány tak, aby řešením této rovnice byla chaotická časová řada. Celkem bylo vygenerováno 701 vzorků. Data v diskrétní časový okamžiku $k = 523$ pak byly perturbovány podle následujícího předpisu.

$$y(523) = y(523) + 0.05 \cdot y(523) \quad (6.2)$$

Výsledná časová řada a detail pertubace je znázorněn na obrázku 6.1. Jako adaptivní filtr byl zvolen QNU (resp. Volterrův filtr) jehož vstupem jsou 4 časově spožděné hodnoty časové řady $y(k-1)$, $y(k-2)$, $y(k-3)$, $y(k-4)$. Hodnoty adaptivních vah tohoto filtru byly adaptovány algoritmem NLMS. Struktura filtru byla zvolena stejně jako v [111]. Rychlost učení během experimentu byla nastavena na $\mu = 1$. Metoda POT byla zvolena podle rovnice 4.5 a délka okna pro detekci novosti algoritmem ESE byla nastavena na $n_s = 300$.

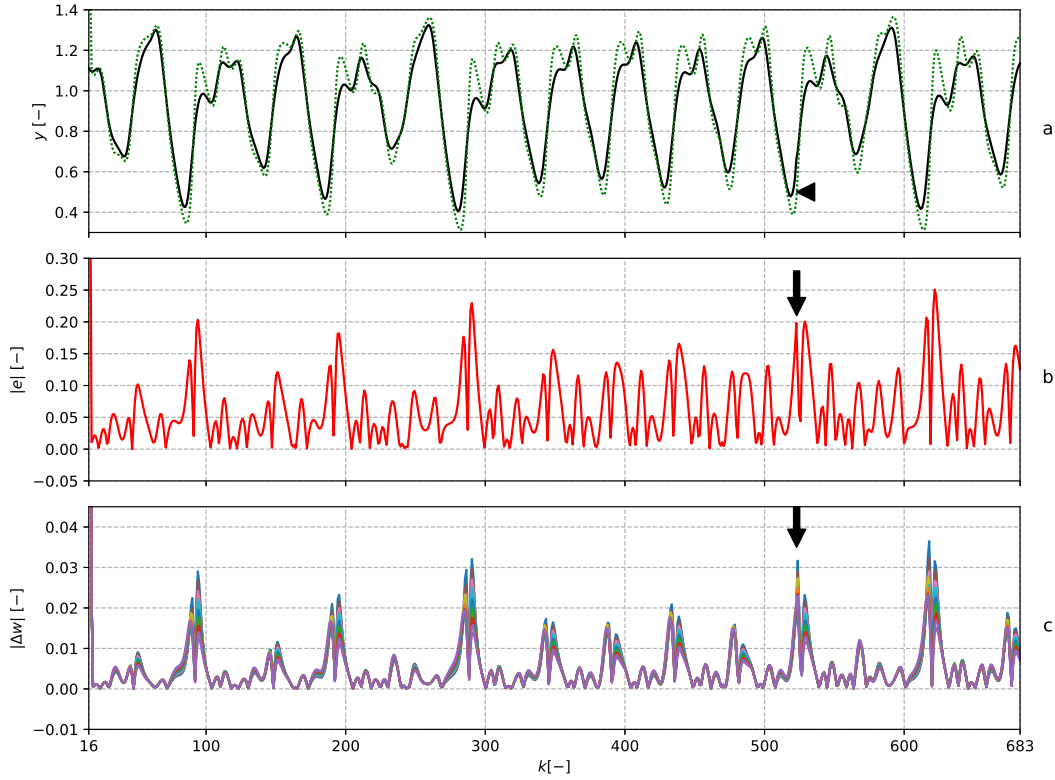
Výstup adaptivního filtru a chyba predikce jsou znázorněny na obrázku 6.2. Výsledky metod detekce novosti jsou zobrazeny na obrázku 6.3. Z obrázku je patrné, že globální maximum průběhu *ESE* odpovídá pertubovaným datům. Globální maximum metod *ELBND* a *LE* odpovídá vzorku u něhož byla největší chyba predikce. Hodnoty prahů algoritmu LE byly zvoleny $\alpha = \{4, 5, 6, 7, 8, 9\}$ a délka okna byla nastavena na $m = 30$.



Obrázek 6.1: Horní graf zobrazuje celou datovou řadu. Spodní grafy zobrazují detail pertubovaného vzorku v diskrétní časový okamžik $k = 523$.

6.2 Detekce změny rozptylu šumu v náhodném datovém toku

Tato případová studie je navržena na základě problému, který se vyskytuje v použití hybridních navigačních systémů využívajících GPS (Global Positioning System) senzory pro navigaci výpočtem [136]. Smyslem experimentu je demonstrovat možnost využití algoritmu ESE pro



Obrázek 6.2: Graf (a) zobrazuje datovou řadu s pertubací (černá plná čára) a výstup adaptivního filtru (tečkovaná zelená čára). Pertubovaný vzorek je označen černou šipkou. Graf (b) zobrazuje velikost chyby predikce e (resp. její absolutní hodnotu). Na grafu (c) jsou znázorněny přírůstky adaptivních vah filtru (resp. absolutní hodnotu těchto přírůstků).

detekci změn rozptylu šumu v náhodných datech.

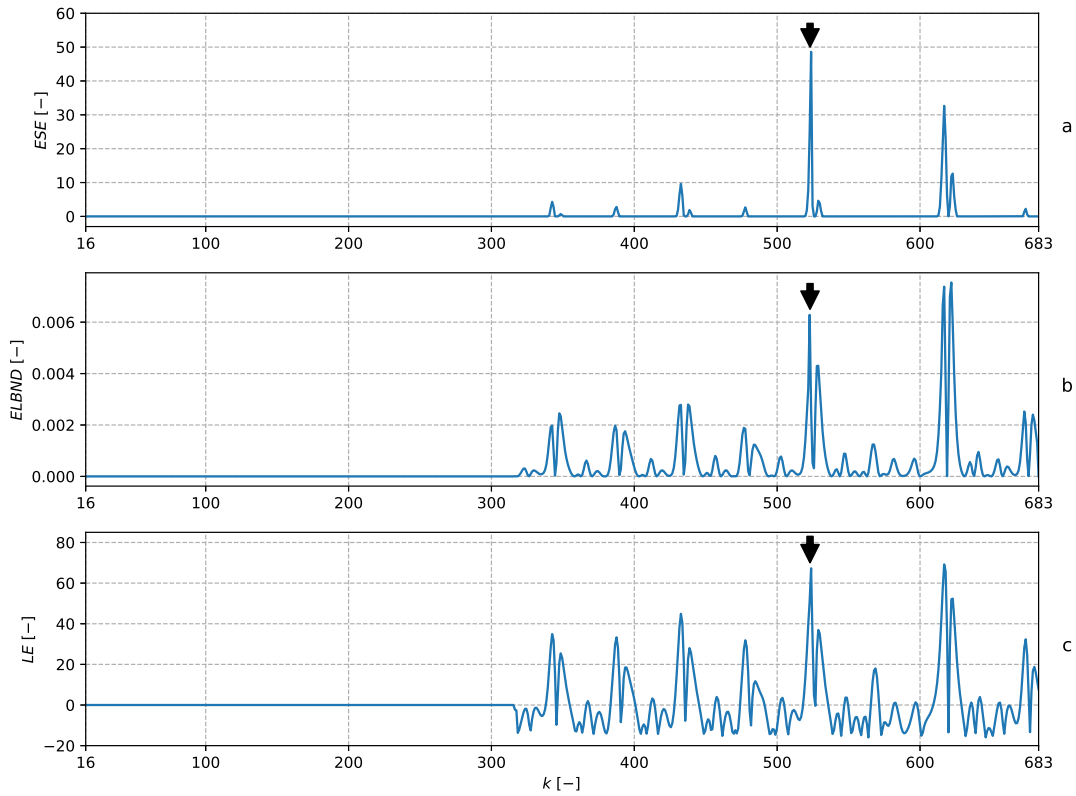
Uvažujme dva vstupy $x_1(k)$ a $x_2(k)$ a výstup generátoru signálu $y(k)$ takový, že

$$y(k) = x_1(k) + x_2(k) + x_1(k) \cdot x_2(k) + v(k) \quad (6.3)$$

kde člen $v(k)$ reprezentuje aditivní Gaussovský šum který je přidán k výstupu generátoru $y(k)$. Přidaný šum má nulovou střední hodnotu a směrodatnou odchylku $\sigma_n = 0.1$, takže $v(k) \sim N(0, 1)$. Hodnoty vstupů jsou v každém diskrétním časovém okamžiku vybrány náhodně z rovnoměrného rozdělení na intervalu $\langle 0, 1 \rangle$. V diskrétním časovém okamžiku $k = 500$ dojde ke změně směrodatné odchylky šumu na hodnotu $\sigma_n = 0.2$, $v(k) \sim N(0, 0.2)$. Adaptivní filtr v tomto experimentu byl QNU ve tvaru

$$\hat{y}(k) = w_1 \cdot x_1(k) + w_2 \cdot x_2(k) + w_3 \cdot x_1(k) \cdot x_2(k) \quad (6.4)$$

tak, že jeho struktura odpovídá struktuře generátoru signálu. Adaptivní parametry filtru byly adaptovány algoritmem GNGD. Rychlost učení byla nastavená jako $\mu = 1$. Metoda POT



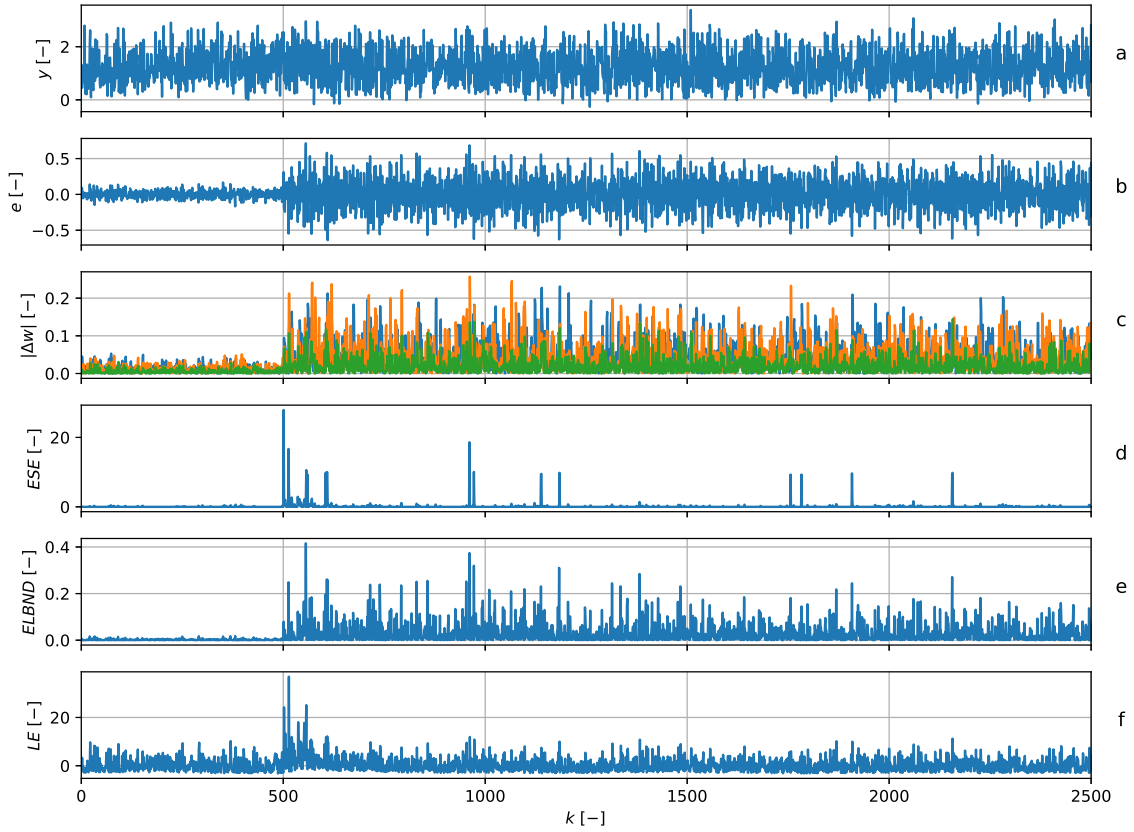
Obrázek 6.3: Graf (a) zobrazuje hodnotu ESE. Prvních 300 vzorků je hodnota ESE nulová, protože délka okna pro vyhodnocování novosti $n_s = 300$. Graf (b) zobrazuje výsledky algoritmu ELBND. Prvních 300 výsledků ELBND je pro názornost vynecháno. Graf (c) zobrazuje výsledky algoritmu LE.

byla zvolena podle rovnice 4.6 a délka okna $n_s = 500$. Výsledky experimentu jsou zobrazeny na obrázku 6.4. Apriorní hodnoty parametrů GPD byly stanoveny na základě 500 vzorků, které nejsou v následujícím obrázku 6.4 zobrazeny. Globální maximum ESE odpovídá změně směrodatné odchylky šumu σ_n . Detekce pomocí algoritmů LE a ELBND je o několik vzorků opožděná. Pro výpočet LE byl použit vztah 3.15 a délka okna byla nastavena na $M = 300$. Výpočet ELBND byl proveden podle rovnice 3.18.

6.3 Detekce skokové změny parametrů generátoru signálu

Tato případová studie je motivována problémem, který vzniká při sledování vícero náhodných datových toků [137] u kterých se kontroluje, zda nedošlo ke změně vlastností jejich generátoru. Uvažujme opět dva vstupy $x_1(k)$, $x_2(k)$ a výstup generátoru signálu ve tvaru

$$y(k) = x_1(k) + x_2(k) + x_1(k) \cdot x_2(k) + v(k) \quad (6.5)$$



Obrázek 6.4: Detekce změny rozptylu šumu. Na grafu (a) jsou zobrazeny data z generátoru (modrá) a výstup z adaptivního filtru (zelená). Na grafu (b) je vynesena chyba filtru $e(k)$. Graf (c) zobrazuje velikosti přírůstků adaptivních parametrů filtru. Na grafu (d) jsou zobrazeny hodnoty ESE. V diskrétní časový okamžik $k = 500$ je patrný značný nárůst v ESE, který reflektuje změnu směrodatné odchylky aditivního šumu. Na dalších grafech (e) a (f) jsou zobrazeny výsledné hodnoty algoritmů ELBND a LE.

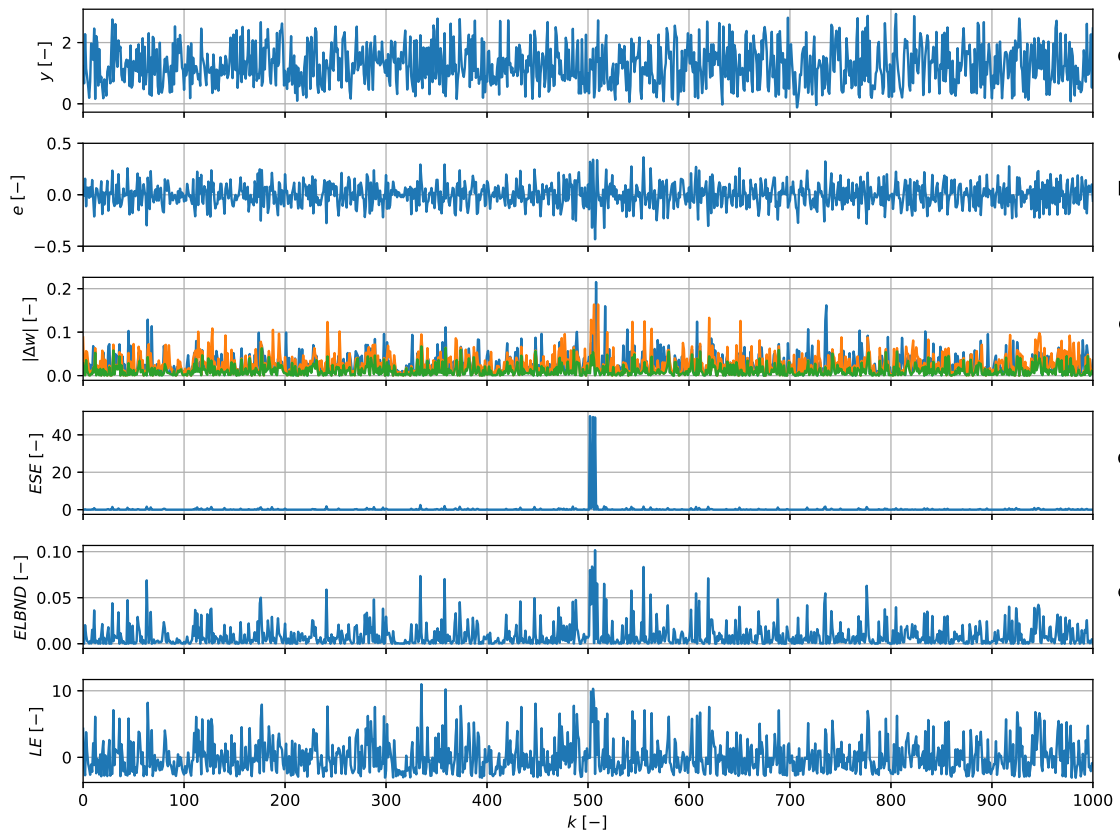
kde člen $v(k)$ reprezentuje gaussovský aditivní šum s nulovou střední hodnotou a směrodatnou odchylkou $\sigma_n = 0.1$, $v \sim N(0, 0.1)$. Hodnoty vstupů $x_1(k)$ a $x_2(k)$ jsou v každém diskrétním časovém okamžiku náhodně vybrány z rovnoměrného rozdělení, $x_1(k) \sim U(0, 1)$ resp. $x_2(k) \sim U(0, 1)$. V časový diskrétní okamžik $k = 500$ dojde ke změně parametrů generátoru, a výstup generátoru přejde do tvaru

$$y(k) = 0.4 \cdot x_1(k) + 1.6x_2(k) + 0.99x_1(k) \cdot x_2(k) + v(k). \quad (6.6)$$

Jako adaptivní filtr byl v tomto případě zvolen QNU, jehož struktura odpovídá generátoru dat. Výstup tohoto filtru je tedy

$$\hat{y}(k) = w_1 \cdot x_1(k) + w_2 \cdot x_2(k) + w_3 \cdot x_1(k) \cdot x_2(k) \quad (6.7)$$

přičemž adaptivní parametry uvedeného filtru jsou adaptovány algoritmem GN GD. Rychlost učení během experimentu byla nastavena na $\mu = 1$. POT metoda byla zvolena podle rovnice 4.5 a parametr délky okna byl $n_s = 500$. Apriorní hodnota parametrů GPD a dat pro LE byla získána použitím 500 vzorků dat (vygenerovaných podle rovnice 6.6). Výsledky experimentu jsou znázorněny na obrázku 6.5. Z obrázku je patrné, že pomocí algoritmu ESE se podařilo detekovat skokovou změnu parametrů, čemuž odpovídá výrazné globální maximum v ESE. Globální maximum v LE neodpovídá skokové změně parametrů generátoru a detekce pomocí algoritmu ELBND je opožděná.



Obrázek 6.5: Detekce skokové změny generátoru signálu. Na grafu (a) je zobrazena původní časová řada (modrá). Graf (b) zobrazuje chybu filtru e . Na grafu (c) jsou zobrazeny velikosti přírůstků adaptivních vah filtru. Na grafu (d) jsou pak výsledky algoritmu ESE, přičemž k skokové změně parametrů generátoru signálu došlo v diskrétní časový okamžik $k = 500$. Je tedy vidět globální maximum v ESE odpovídající úspěšné detekci. Na grafech (e) a (f) jsou pak výsledky metod ELBND a LE. Detekci algoritmem ELBND lze považovat za úspěšnou. Detekce algoritmem LE byla neúspěšná. Globální maximum LE je v diskrétním časovém okamžiku $k = 338$, který neodpovídá skokové změně parametrů generátoru signálu.

6.4 Detekce náhlé absence šumu

V této kapitole je ukázáno, že lehce modifikovaný algoritmus ESE může být využit také k detekci neobvykle malých změn parametrů adaptivního filtru. Oproti standardní variantě ESE budeme vyhodnocovat neobvykle malé přírůstky vah adaptivního filtru. Takže jediná změna v algoritmu je, že metodou POT budeme vybírat pouze nejmenší změny adaptivních vah a budeme odhadovat parametry GPD z takto vybraných hodnot.

Uvažujme dva vstupy $x_1(k)$ a $x_2(k)$ jejichž hodnoty jsou v každém diskrétním časovém okamžiku k vybrány z rovnoměrného rozdělení, takže $x_1(k) \sim U(0, 1)$ a $x_2(k) \sim U(0, 1)$. Výstup generátoru dat $y(k)$ je definován jako

$$y(k) = x_1(k) + x_2(k) + x_1(k) \cdot x_2(k) + v(k) \quad (6.8)$$

kde člen $v(k)$ reprezentuje aditivní gaussovský šum s nulovou střední hodnotou a směrodatnou odchylkou $\sigma_n = 0.1$. V diskrétním časovém okamžiku dojde k odstranění aditivního šumu a výstup generátoru signálu přejde do tvaru

$$y(k) = x_1(k) + x_2(k) + x_1(k) \cdot x_2(k). \quad (6.9)$$

který platí pro všechna $k \geq 500$.

Jako adaptivní filtr byl zvolen QNU, jehož výstup je definován

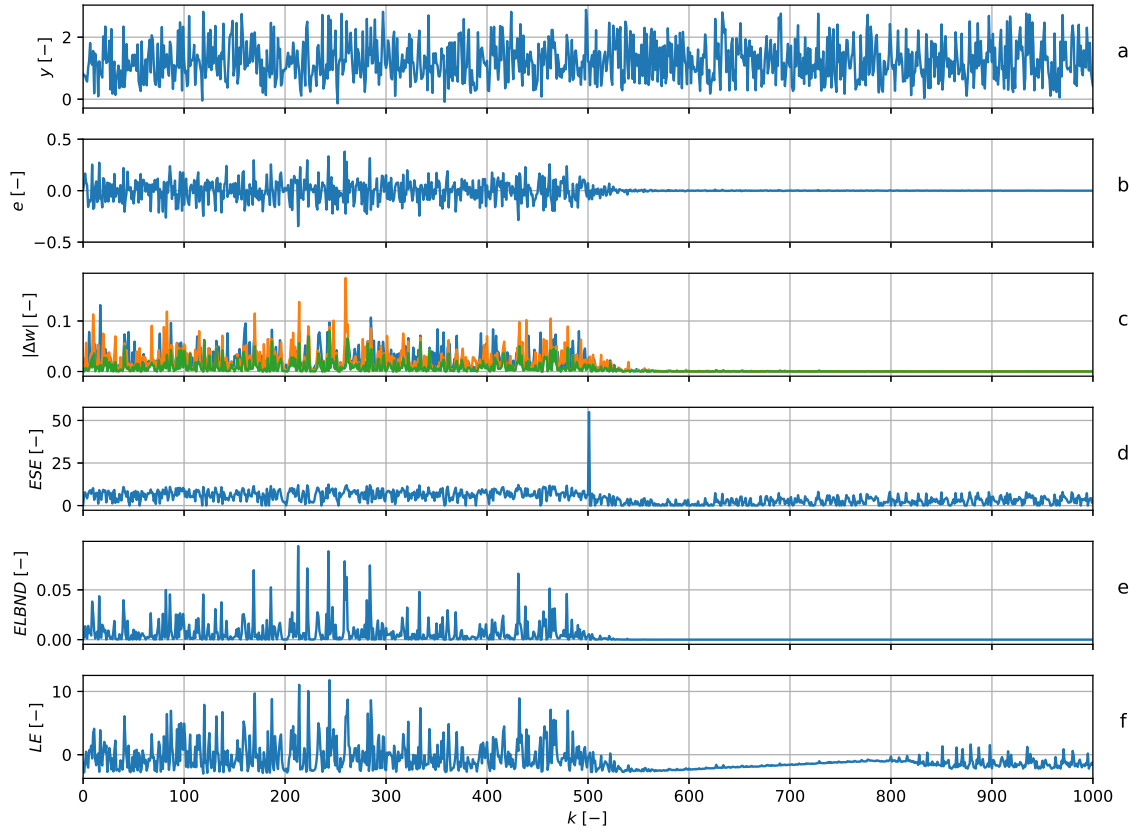
$$\hat{y}(k) = w_1 \cdot x_1(k) + w_2 \cdot x_2(k) + w_3 \cdot x_1(k) \cdot x_2(k) \quad (6.10)$$

takže jeho struktura odpovídá generátoru signálu. Parametry toho filtru jsou adaptovány algoritmem GNGD. Rychlost učení byla nastavena jako $\mu = 1$. Metoda POT byla zvolena podle rovnice 4.5 a parametr $n_s = 500$. Pro výpočet LE byl použit vztah 3.15 a délka okna byla nastavena na $M = 300$. Výpočet ELBND byl proveden podle rovnice 3.18. Na obrázku 6.6 jsou zobrazeny výsledky experimentu. Maximum v ESE odpovídá detekci vymizení šumu z generátoru signálu. Výsledky metod ELBND a LE jsou uvedeny pouze pro ilustraci.

6.5 Detekce změny trendu

Cílem této kapitoly je demonstrovat použití algoritmu ESE při detekci změny trendu, což je úloha, která se často vyskytuje v oblasti detekce poruch a diagnostice [138]. Uvažujme opět dva vstupy $x_1(k) \sim U(0, 1)$ a $x_2(k) \sim U(0, 1)$ a výstup generátoru dat $y(k)$ takový, že

$$y(k) = x_1(k) + x_2(k) + 0.01 \cdot k + v(k) \quad (6.11)$$



Obrázek 6.6: Detekce vymizení šumu ze signálu. Na grafu (a) je zobrazena původní časová řada (modrá). Graf (b) zobrazuje chybu filtru e . Na grafu (c) jsou zobrazeny velikosti přírůstků adaptivních vah filtru. Na grafu (d) jsou pak výsledky modifikovaného algoritmu ESE, přičemž k odstranění šumu ze signálu došlo v diskretní časový okamžik $k = 500$. Je tedy vidět globální maximum v ESE odpovídající úspěšné detekci vymizení šumu ze signálu. Na grafech (e) a (f) jsou pak výsledky získané pomocí metod ELBND a LE.

kde člen $v(k)$ reprezentuje aditivní gaussovský šum s nulovou střední hodnotou a směrodatnou odchylkou $\sigma_n = 0.1$. V diskretním časovém okamžiku $k = 500$ nastane změna trendu. Výstup generátoru signálu se změní, tak, že

$$y(k) = x_1(k) + x_2(k) + 0.0105 \cdot k + v(k), \quad (6.12)$$

pro $k \geq 500$.

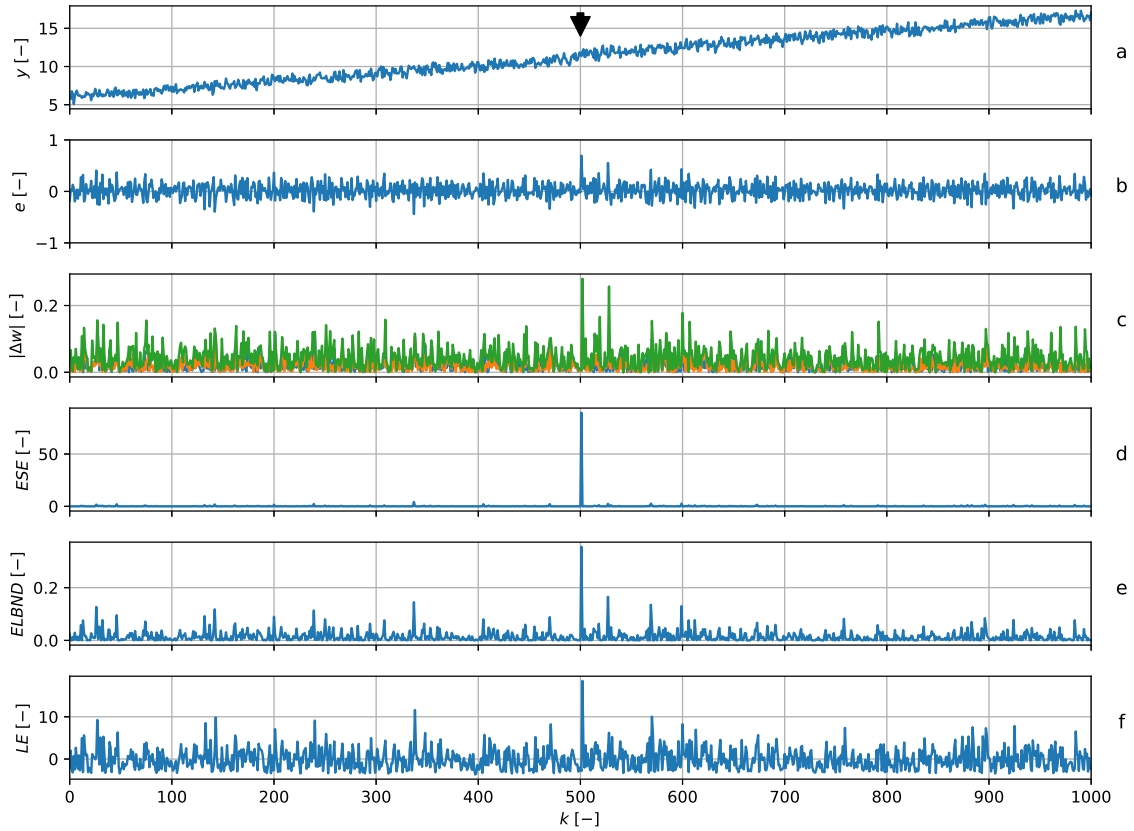
Pro zpracování signálu byl použit filtr typu LNU s třemi vstupy, takže výstup uvedeného filtru je ve tvaru

$$\hat{y}(k) = w_1 \cdot x_1(k) + w_2 \cdot x_2(k) + w_3 \quad (6.13)$$

takže odpovídající vektor vstupů je

$$\mathbf{x}(k) = [x_1(k), x_2(k), 1]. \quad (6.14)$$

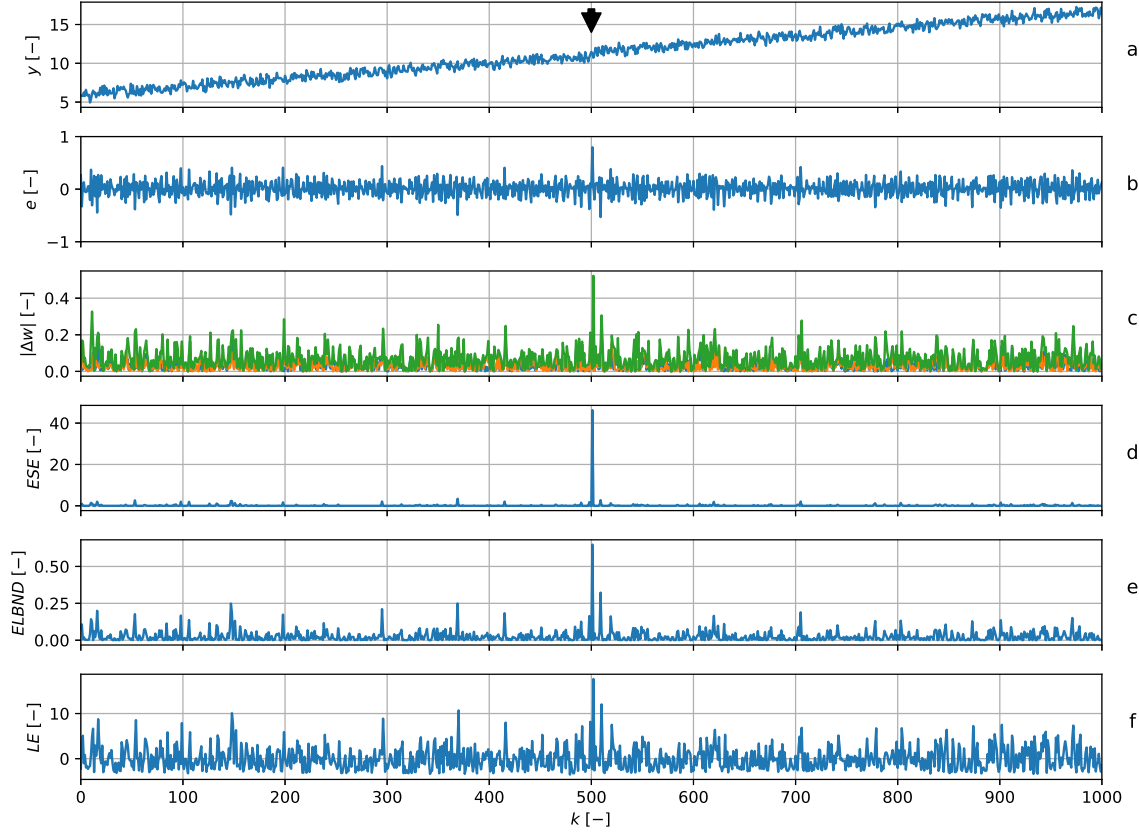
Struktura LNU byla vybrána tak, aby co nejlépe odpovídala struktuře generátoru signálu. Parametry adaptivního filtru byly v tomto experimentu adaptovány algoritmem GNGD. Rychlost učení byla nastavena na $\mu = 1$, délka okna byla $n_s = 500$ a POT metoda byla zvolena podle 4.5. Na obrázku 6.7 jsou zobrazeny výsledky experimentu. Globální maximum ESE odpovídá okamžiku změny trendu. Algoritmy LE a ELBND úspěšně změnu trendu také detekovali. Pro výpočet LE byl použit vztah 3.15 a délka okna byla nastavena na $M = 300$. Výpočet ELBND byl proveden podle rovnice 3.18.



Obrázek 6.7: Detekce změny trendu při použití algoritmu GNGD. Na grafu (a) jsou zobrazena data z generátoru signálu (modré). Černá šipka znázorňuje okamžik ve kterém došlo ke změně trendu. Na grafu (b) je zobrazena chyba adaptivního filtru e . Na grafu (c) jsou znázorněny velikosti přírůstků adaptivních vah filtru. Grafy (d), (e) a (f) znázorňují výsledky detekce novosti pomocí algoritmů ESE, ELBND a LE. Všechny tři algoritmy vykazují úspěšnou detekci změny trendu, která koresponduje s jejich maximální hodnoty během experimentu.

Vzhledem k tomu, že pro algoritmus ESE není důležité, jaký konkrétní adaptivní algoritmus je použit, jsou na následujících obrázcích ještě zobrazeny výsledky detekce změny trendu

s použitím algoritmů RLS (viz obrázek 6.8) a LMS (viz obrázek 6.9). Pro algoritmus RLS bylo prvotní nastavení parametru $\delta = 10$, pro algoritmus LMS byla nastavena rychlost učení $\mu = 0.1$. Výsledky těchto dvou algoritmů publikovány nebyly.

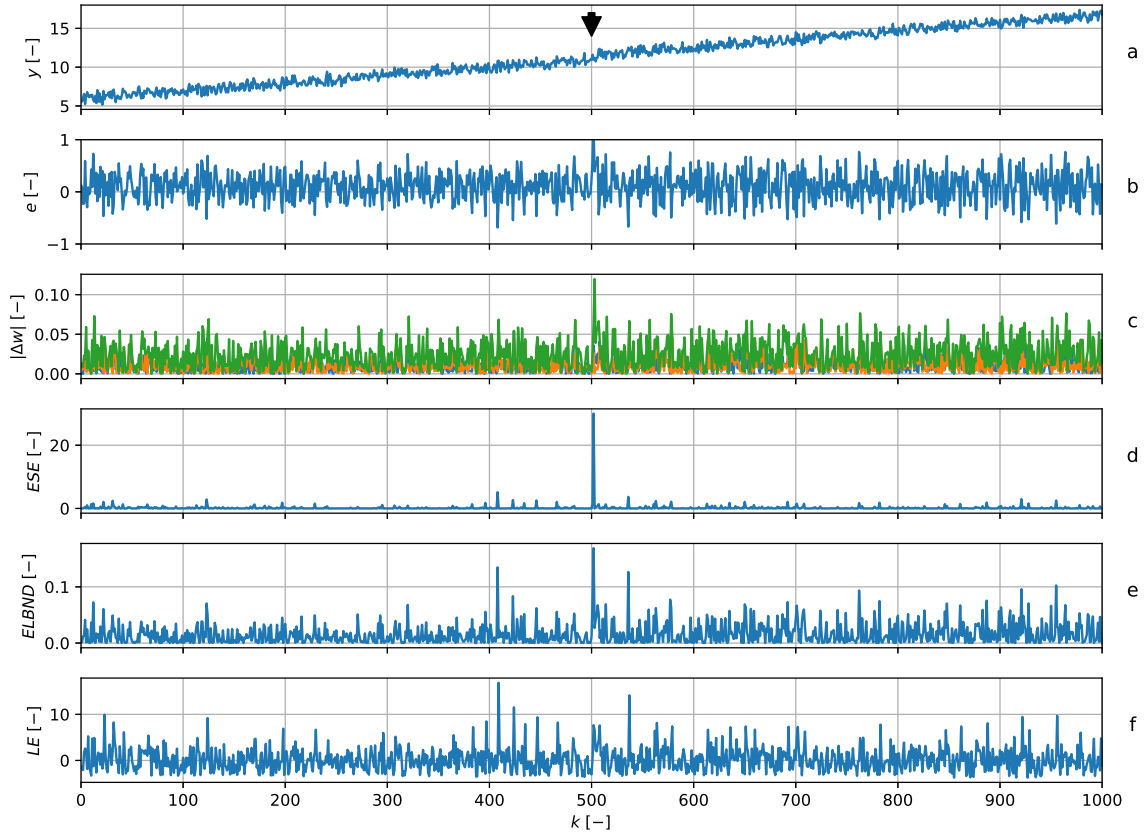


Obrázek 6.8: Detekce změny trendu při použití algoritmu RLS. Na grafu (a) jsou zobrazena data z generátoru signálu (modré). Černá šipka znázorňuje okamžik ve kterém došlo ke změně trendu. Na grafu (b) je zobrazena chyba adaptivního filtru e . Na grafu (c) jsou znázorněny velikosti přírůstků adaptivních vah filtru. Grafy (d), (e) a (f) znázorňují výsledky detekce novosti pomocí algoritmů ESE, ELBND a LE. Všechny tři algoritmy vykazují úspěšnou detekci změny trendu, která koresponduje s jejich maximální hodnoty během experimentu.

6.6 Detekce epilepsie v EEG záznamu myši

Poslední případová studie je věnována detekci epileptického záchvatu v signálu EEG myši pomocí algoritmu EEG. Standartizovaná data ze tří vybraných kanálů EEG, ve kterých byl expertem stanoven začátek epileptického záchvatu přibližně v čase $k \approx 1700$, jsou zobrazeny na obrázku 6.10. Standartizace byla provedena podle předpisu

$$y = \frac{x - \mu_x}{\sigma_x} \quad (6.15)$$



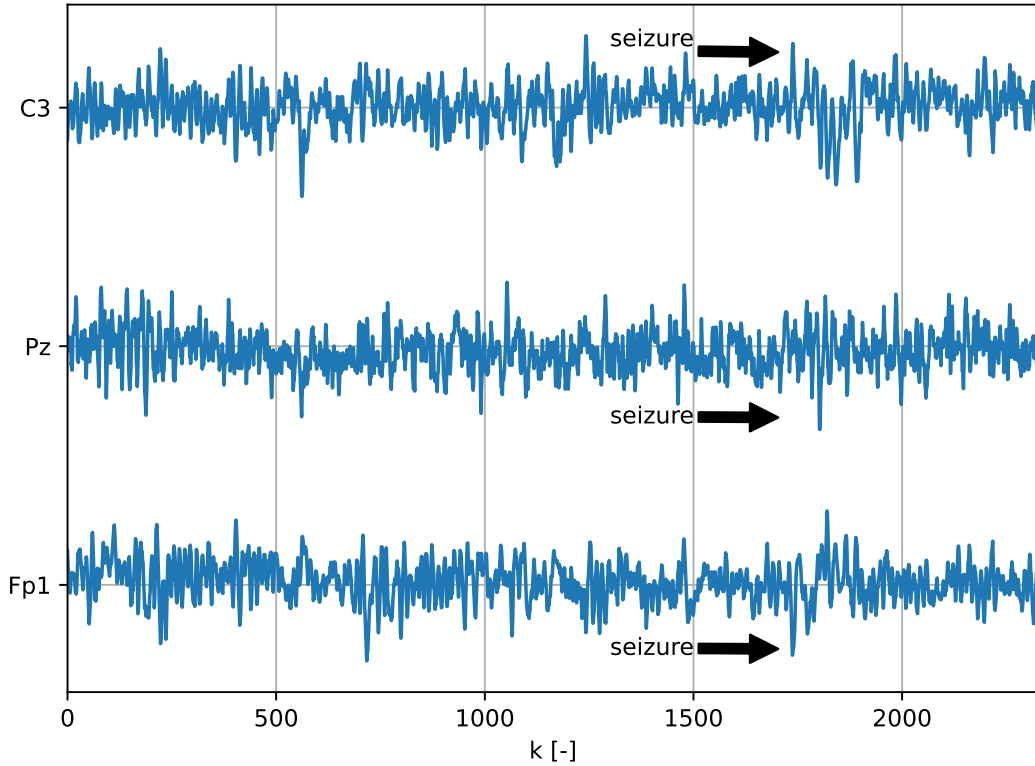
Obrázek 6.9: Detekce změny trendu při použití algoritmu LMS. Na grafu (a) jsou zobrazena data z generátoru signálu (modré). Černá šipka znázorňuje okamžik ve kterém došlo ke změně trendu. Na grafu (b) je zobrazena chyba adaptivního filtru e . Na grafu (c) jsou znázorněny velikosti přírůstků adaptivních vah filtru. Grafy (d), (e) a (f) znázorňují výsledky detekce novosti pomocí algoritmů ESE, ELBND a LE. Všechny tři algoritmy vykazují úspěšnou detekci změny trendu, která koresponduje s jejich maximální hodnoty během experimentu.

kde y je výsledná standartizovaná hodnota, x je původní hodnota, μ_x je průměrná hodnota původních dat daného kanálu a σ_x je jejich původní směrodatná odchylka.

Jako adaptivní filtr byl, na základě experimentů, zvolen FIR filtr délky 10. Vstupem je vektor dat

$$\mathbf{x} = [x(k-1), x(k-2), \dots, x(k-10)] \quad (6.16)$$

takže filtr má 10 adaptivních parametrů. Filtr byl adaptován algoritmem NLMS. Rychlost učení byla během experimentu nastavena na $\mu = 1$. Metoda POT byla zvolena podle vztahu 4.7 a délka okna $n_s = 1000$. Výsledky detekce algoritmem ESE jsou zobrazeny na obrázku 6.11. Pozice globálního maxima ESE v kanálu C3 (obzvláště signifikantní) je v diskretním časovém okamžiku $k = 1735$, v kanálu Pz je to $k = 1698$ a v kanálu Fp1 je to v $k = 1727$.



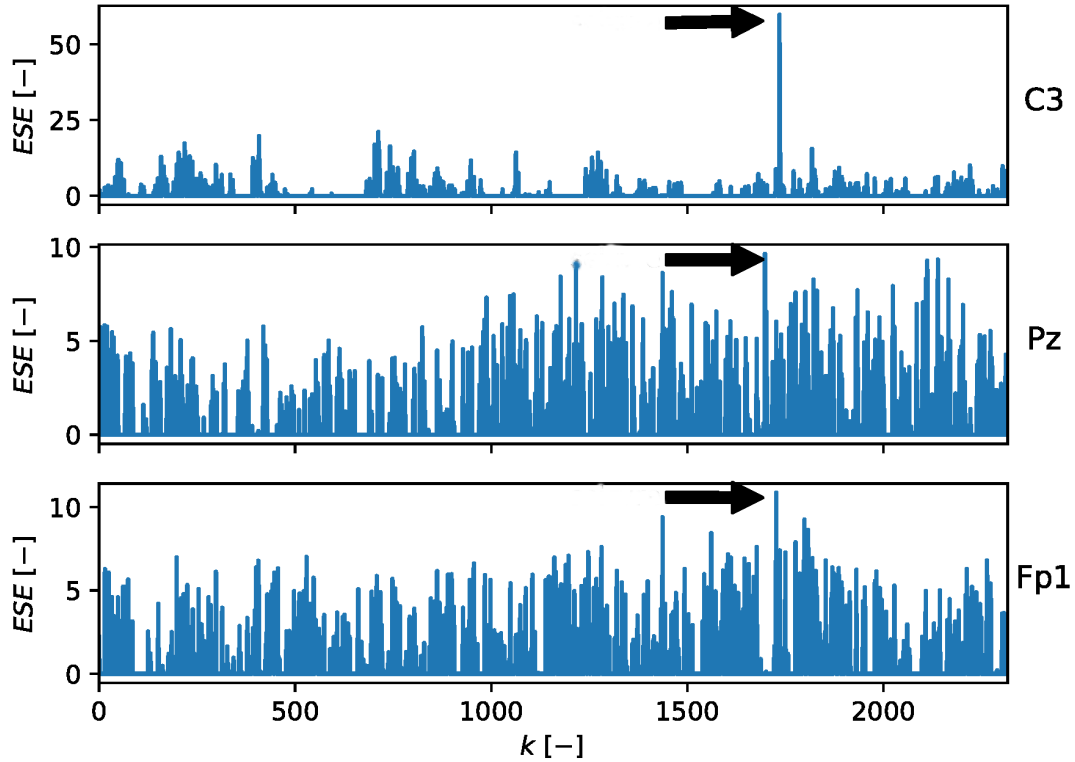
Obrázek 6.10: Vybrané kanály myšního EEG na kterých je patrný epileptický záchvat. Data byly standartizovány. Začátek záchvatu je přibližně v $k \approx 1700$, což znázorňuje černá šipka.

6.7 Vyhodnocení úspěšnosti detekce skokové změny parametrů generátoru signálu

Pro vyhodnocení úspěšnosti skokové změny parametrů generátoru signálu uvažujme generátor signálu s dvěma vstupy $x_1(k)$ a $x_2(k)$ a výstupem $y(k)$ ve tvaru

$$y(k) = a_1 \cdot x_1(k) + a_2 \cdot x_2(k) + a_3 \cdot x_1(k) \cdot x_2(k) + v(k) \quad (6.17)$$

kde člen $v(k)$ reprezentuje gaussovský aditivní šum s nulovou střední hodnotou a směrodatnou odchylkou σ . Počáteční hodnoty parametrů a_1 , a_2 a a_3 jsou vygenerovány z rovnoměrného rozdělení $U(-1, 1)$. V diskrétním časovém okamžiku $k = 200$, dojde ke skokové změně těchto parametrů a jejich nová hodnota je opět náhodně vygenerována z rovnoměrného rozdělení $U(-1, 1)$. Celkový počet vzorků experimentu je 400. Použitý adaptivní filtr je stejný jako v předchozí případové studii detekce skokové změny parametrů generátoru signálu, viz kapitola 6.3. Parametry tohoto adaptivního filtru byly adaptovány algoritmem GNGD. Metoda POT byla zvolena podle rovnice 4.5 a délka okna byla zvolena $n_s = 1200$. Apriorní informace o parametrech GPD byla pro každý experiment získána pomocí 1200 vzorků, s počátečními



Obrázek 6.11: Hodnota ESE pro vybrané kanály se záznamem myšího EEG ve kterých je patrný epileptický záchvat. V kanále C3 je v ESE výrazný nárůst po začátku záchvatu (přibližně v $k \approx 1700$), v porovnání s ostatními kanály. Černá šipka znázorňuje přibližný začátek epileptického záchvatu.

hodnotami parametrů a_1 , a_2 a a_3 . Pro každý experiment byla vyhodnocena hodnota SNR jako

$$SNR = 10 \log_{10} \frac{\sigma_s^2}{\sigma_n^2} \quad (6.18)$$

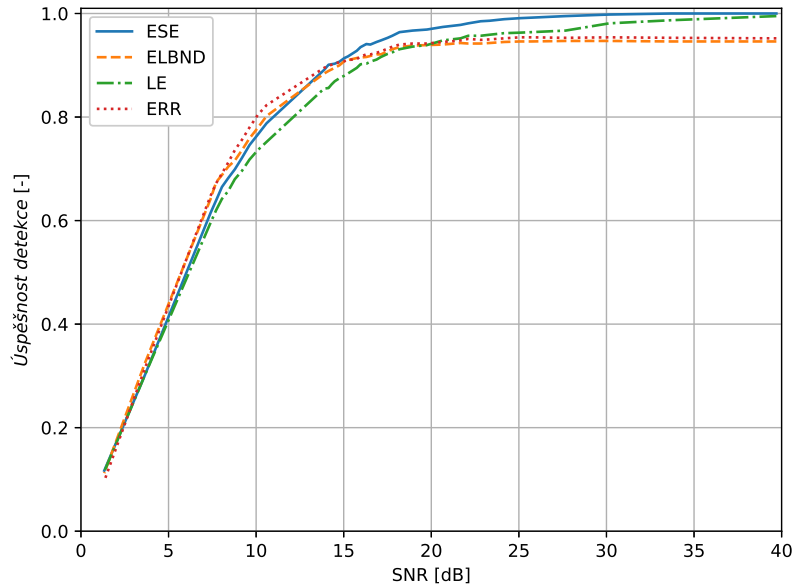
kde σ_s je hodnota směrodatné odchylky výstupu generátoru signálu během experimentu a σ_n je směrodatná odchylka aditivního gaussovského šumu. Vyhodnocení přesnosti detekce bylo provedeno následujícím způsobem:

1. nastavení hodnoty směrodatné odchylky šumu σ_n
2. pro zvolenou hodnotu směrodatné odchylky σ_n se provede 1000 experimentů, přičemž pro každý experiment jsou nově vygenerovány počáteční hodnoty parametrů generátoru signálu a_1 , a_2 a a_3 .
3. pro každý experiment je vyhodnocena úspěšnost detekce. Za úspěšnou detekci je považováno, pokud globální maximum ESE, ELBND, EL respektive chyby filtru je v mezích

$$k \geq 200 \text{ a } k \leq 210.$$

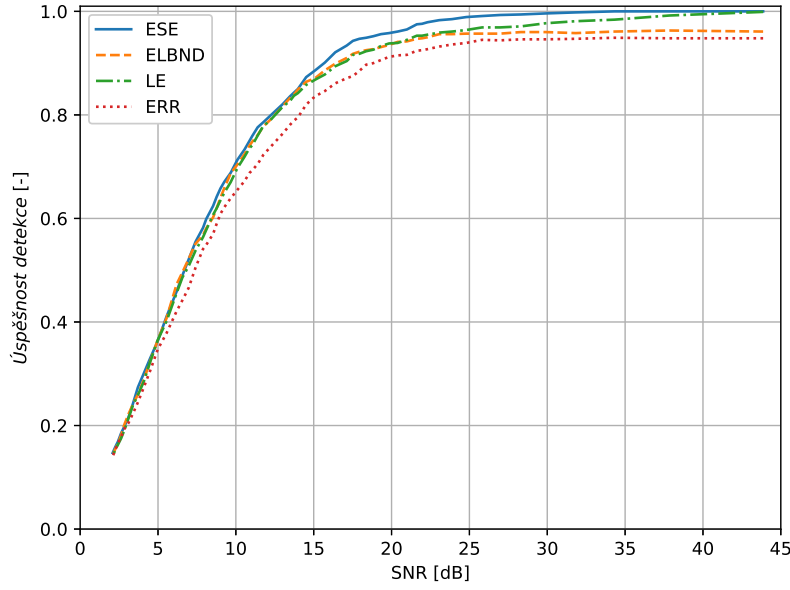
4. vypočte se celková úspěšnost detekce pro danou hodnotu směrodatné odchylky (poměr počtu úspěšných detekcí k celkovému počtu experimentů)
5. pro každý experiment se vyhodnotí SNR podle 6.18 a pak se pro zvolenou hodnotu σ vypočítá průměrná hodnota SNR pro všechny experimenty

Vyhodnocení úspěšnosti detekce bylo vyhodnoceno pro dva případy. V prvním případě, byly hodnoty vstupů $x_1(k)$ a $x_2(k)$ generovány z rovnoměrného rozdělení $U(-1, 1)$. Výsledky úspěšnosti detekce pro různé hodnoty směrodatných odchylek šumu σ_n jsou zobrazeny na obrázku 6.12. Výsledky jsou také shrnuty v tabulce 2 (viz příloha C). Pro porovnání jsou zvoleny metody ELBND (výpočet podle rovnice 3.18), LE s oknem $n_s = 1200$ (výpočet podle rovnice 3.15) a velikost chyby adaptivního filtru e (v grafu označeno jako ERR).



Obrázek 6.12: Úspěšnost detekce skokové změny parametrů generátoru signálu. Hodnoty vstupů generátoru signálu jsou generovány z rovnoměrného rozdělení $U(-1, 1)$. Pro hodnoty $SNR > 15 \text{ dB}$ dosáhl algoritmus ESE vyšší úspěšnost než algoritmy LE, ELBND a vyhodnocení pomocí chyby filtru (ERR). Pro $SNR > 33 \text{ dB}$ dosáhl algoritmus ESE 100% úspěšnost detekce.

Ve druhém případě byly hodnoty vstupů $x_1(k)$ a $x_2(k)$ generovány z normálního rozdělení. Vyhodnocení úspěšnosti detekce bylo provedeno stejně jako v případě popsáném výše. Výsledky úspěšnosti detekce pro různé hodnoty směrodatných odchylek šumu σ_n jsou zobrazeny na obrázku 6.13. Uvedené výsledky v číselné podobě jsou uvedeny v tabulce 3 (viz příloha C). Výpočet hodnot ELBND, LE a ERR bylo provedeno stejně jako ve výše uvedeném případě.



Obrázek 6.13: Úspěšnost detekce skokové změny parametrů signálu. Hodnoty vstupů generátoru signálu jsou generovány z normálního rozdělení $N(0, 1)$. Pro hodnoty $SNR > 8 \text{ dB}$ dosáhl algoritmus ESE lepší úspěšnosti detekce než algoritmy LE, ELBND a vyhodnocení pomocí velikosti chyby predikce (ERR). Pro $SNR > 34 \text{ dB}$ dosáhl algoritmus ESE 100% úspěšnosti detekce.

6.8 Vyhodnocení úspěšnosti detekce skokové změny trendu

Pro vyhodnocení úspěšnosti změny trendu uvažujme výstup generátoru signálu $y(k)$ se dvěma vstupy $x_1(k)$ a $x_2(k)$ jehož výstup je definován jako

$$y(k) = x_1(k) + x_2(k) + 0.01 \cdot k + v(k) \quad (6.19)$$

kde člen $v(k)$ reprezentuje aditivní gaussovský šum s nulovou střední hodnotou a směrodatnou odchylkou σ_n . V diskretním časovém okamžiku $k = 200$ se změní výstup generátoru signálu

$$y(k) = x_1(k) + x_2(k) + (0.01 + a) \cdot k + v(k) \quad (6.20)$$

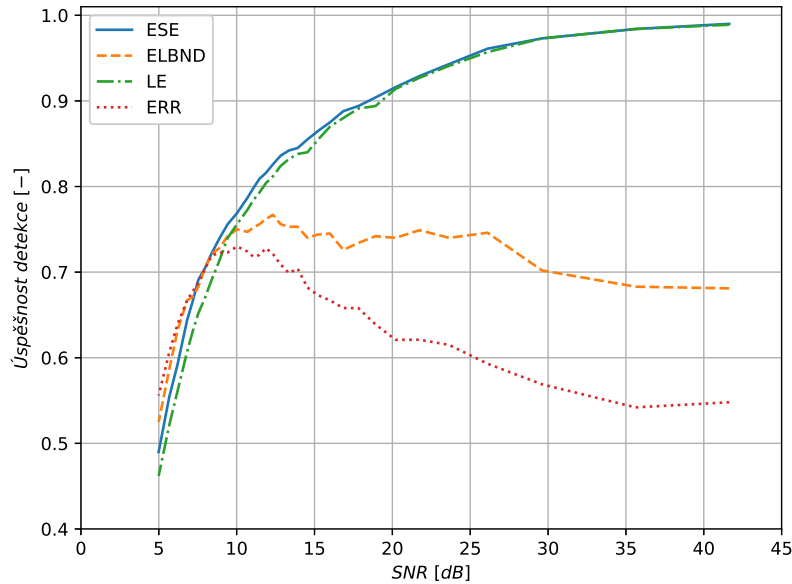
přičemž parametr a je vygenerován v každém experimentu z rovnoměrného rozdělení $U(-0.02, 0.02)$. Počet vzorků experimentu je 400.

Struktura adaptivního filtru byla zvolena stejně jako v předcházející případové studii detekce změny trendu (viz kapitola 6.3). Adaptivní parametry filtru byly adaptovány algoritmem GNGD. Metoda POT pro algoritmus ESE byla zvolena jako 4.5 a délka okna byla nastaven na $n_s = 1200$. Apriorní informace o parametrech GPD byla získána na základě 1200 vzorků, ve kterých nedošlo ke změně trendu.

Vyhodnocení přesnosti detekce změny trendu bylo provedeno následujícím způsobem:

1. nastavení hodnoty směrodatné odchylky šumu σ_n
2. pro zvolenou hodnotu směrodatné odchylky σ_n se provede 1000 experimentů. V každém experimentu dojde v diskrétním časovém okamžiku $k = 200$ k novému vygenerování hodnoty parametru a z rovnoměrného rozdělení $U(-0.02, 0.02)$.
3. pro každý experiment je vyhodnocena úspěšnost detekce. Za úspěšnou detekci je považováno, pokud globální maximum ESE, ELBND, EL respektive chyby filtru je v mezích $k \geq 200$ a $k \leq 210$.
4. vypočte se celková úspěšnost detekce pro danou hodnotu směrodatné odchylky šumu σ_n (poměr počtu úspěšných detekcí k celkovému počtu experimentů)
5. pro každý experiment se vyhodnotí SNR podle 6.18 a pak se pro zvolenou hodnotu σ_n vypočítá průměrná hodnota SNR pro všechny experimenty

Vyhodnocení úspěšnosti detekce změny trendu bylo vyhodnoceno pro hodnoty vstupů $x_1(k)$ a $x_2(k)$ vygenerované z rovnoměrného rozdělení $U(-1, 1)$. Výsledky úspěšnosti detekce pro různé hodnoty směrodatných odchylek šumu σ_n jsou zobrazeny na obrázku 6.14. Výsledky jsou také shrnuty v tabulce 1 (viz příloha C). Pro porovnání jsou zvoleny metody ELBND (výpočet podle rovnice 3.18), LE s oknem $n_s = 1200$ (výpočet podle rovnice 3.15) a velikost chyby adaptivního filtru e (v grafu označeno jako ERR).



Obrázek 6.14: Úspěšnost detekce změny trendu. Hodnoty vstupů generátoru signálu byly generovány z rovnoměrného rozdělení $U(-1, 1)$. Pro hodnoty $SNR > 8 \text{ dB}$ dosáhl algoritmus ESE větší úspěšnosti detekce než LE, ELBND a vyhodnocení pomocí velikosti chyby filtru.

6.9 Evaluace ROC křivky pro detekci změny trendu

Protože úspěšná detekce novosti pomocí algoritmu ESE je závislá na volbě hodnoty, od které budeme považovat hodnotu ESE za „novost“, byl proveden experiment detekce změny trendu a vyhodnocena ROC (Receiver Operating Characteristics) křivka [139]. ROC křivka poskytuje vhodný způsob jak vizualizovat schopnost binárního klasifikátoru klasifikovat správně data na základě proměnlivé velikosti prahu, který klasifikaci určuje (v případě algoritmu ESE je to hodnota ESE) a zároveň umožňuje objektivně jednotlivé klasifikátory porovnávat [140] (více viz následující podkapitola 6.9.2). Pro porovnání algoritmu ESE byly opět zvoleny algoritmy LE, ELBND a klasifikátor, který klasifikuje vzorky náhodně.

Výsledky uvedené v této kapitole byli publikovány v [V3].

6.9.1 Popis experimentu

Stejně jako v případě vyhodnocení přesnosti detekce změny trendu (viz podkapitola 6.8) i v tomto experimentu uvažujeme dva vstupy $x_1(k)$ a $x_2(k)$ výstup generátoru signálu $y(k)$ ve tvaru

$$d(k) = x_1(k) + x_2(x) + 0.01 \cdot k + v(k) \quad (6.21)$$

$$0 \leq k < 200$$

kde člen $v(k)$ reprezentuje gaussovský aditivní šum s nulovou střední hodnotou a směrodatnou odchylkou σ_n . V diskretním časovém okamžiku $k = 200$ přejde výstup generátoru signálu do tvaru

$$y(k) = x_1(k) + x_2(x) + (0.01 + a) \cdot k + v(k) \quad (6.22)$$

$$200 \leq k \leq 399$$

přičemž hodnota parametru a je vygenerována z rovnoměrného rozdělení $U(-0.02, 0.02)$ a pro všechna $200 \leq k \leq 399$ je během daného experimentu konstantní. Hodnoty vstupů $x_1(k)$ a $x_2(k)$ jsou generovány z rovnoměrného rozdělení $U(-1, 1)$.

Jako adaptivní filtr byl zvolen QNU, jehož struktura odpovídá struktuře generátoru signálu. Výstup adaptivního filtru je ve tvaru

$$\hat{y}(k) = w_1 \cdot x_1(k) + w_2 \cdot x_2(k) + w_3 \cdot x_1(k) \cdot x_2(k) \quad (6.23)$$

a parametry toho adaptivního filtru byly adaptovány algoritmem GNGD. Rychlost učení během experimentů byla nastavena na $\mu = 0.5$.

Apriorní hodnota parametrů GPD pro algoritmus ESE je získána pomocí 1200 vzorků, získaných z výstupu generátoru signálu, který je dán rovnicí 6.21. Během experimentů byla délka okna $n_s = 1200$. Metoda POT byla zvolena podle 4.5. Pro výpočet ELBND byl použit vztah

3.18. Výsledky algoritmu LE byly získány pro okno délky $M = 1200$ pomocí vztahu 3.15. Pro každou hodnotu σ bylo provedeno 10000 experimentů na jejichž základě byla zkonstruována ROC křivka. Hodnoty směrodatných odchylek σ_n byly vybrány takto:

$$\sigma_n = \{0.1, 0.2, 0.5, 1.0, 2.0, 2.5\} \quad (6.24)$$

a pro každou hodnotu σ_n byla pro všech 10000 experimentů určená průměrná hodnota SNR , která byla pro každý experiment vypočtena podle rovnice 6.18.

6.9.2 Konstrukce ROC křivky

Pro konstrukci ROC křivky je důležité, aby množina výsledků byla vyvážená. Tedy aby obsahovala stejný počet pozitivních i negativních vzorků. Pro získání vyvážené množiny výsledků byl nejdřív každý experiment převzorkován podle následujícího předpisu

$$ND_r(i) = \max\{ND(i \cdot 10), ND(i \cdot 10 + 1), ND(i \cdot 10 + 2), \dots, ND(i \cdot 10 + 9)\} \quad (6.25)$$

$$i = 0, 1, \dots, 39$$

kde ND reprezentuje hodnotu detektoru novosti (resp. hodnoty algoritmu ESE, ELBND, LE). Z každé převzorkované datové řady jsou vygenerovány dvě množiny. Množina P obsahuje pozitivní vzorek, takový, že $P = \{ND_r(20)\}$ (protože v diskrétním časovém okamžiku $k = 200$ došlo ke změně trendu). Množina N obsahuje zbylých 39 negativních vzorků, takže $N = \{ND(0), \dots, ND(19), ND(21), \dots, ND(39)\}$. Při konstrukci ROC křivky jsou pro každý experiment vybrány dva vzorky. Jeden vzorek z množiny P a jeden náhodně vybraný vzorek z množiny N . Pro vyhodnocení ROC je klíčové zjistit, jestli jsou pozitivní vzorky (prvky množiny P) pro daný práh správně klasifikovány jako pozitivní (True Positive) a zda-li jsou negativní vzorky klasifikovány jako falešně pozitivní (False Positive). Pro danou velikost prahu se určí úspěšnost detekce skutečně pozitivních (True Positive Rate) jako

$$TPR = \frac{TP}{P} = \frac{TP}{10000} \quad (6.26)$$

kde TP je počet správně pozitivně klasifikovaných vzorků a P je celkový počet skutečně pozitivních vzorků. Dále je potřeba určit poměr falešně pozitivních (False Positive Rate) vzorků pro daný práh jako

$$FPR = \frac{FP}{N} = \frac{FP}{10000} \quad (6.27)$$

kde FP je počet vzorků klasifikovaných jako falešně pozitivní a N je celkový počet skutečně negativních vzorků. ROC pak zobrazuje závislost úspěšnost detekce skutečně pozitivních vzorků (TPR) v závislosti na poměru falešně pozitivních vzorků (FPR).

Hodnota TPR bývá nazývána také jako sensitivita. Komplementární hodnotou k FPR je

potom specifita (True Negative Rate TNR), která určuje, kolik opravdu negativních vzorku (TN) je klasifikováno jako negativní. Komplementární ve smyslu

$$TNR = \frac{TN}{N} = 1 - FPR. \quad (6.28)$$

Komplementární k sensitivitě je hodnota míry falešně negativních (FNR), která určuje poměr falešně negativních k (FN) k celkovému počtu pozitivních, tedy

$$FNR = \frac{FN}{P} = 1 - TPR. \quad (6.29)$$

Pro každou ROC křivku je možné určit plochu pod touto křivkou AUROC (Area Under ROC), která vypovídá o schopnosti klasifikátoru rozlišovat mezi jednotlivými třídami. Čím větší plocha pod křivkou, tím víc klasifikátor správně klasifikuje pozitivní případy jako pozitivní a negativní případy jako negativní. Plocha AUROC ideálního klasifikátoru bude 1, zatímco plocha nejhoršího možného klasifikátoru bude rovna 0 (tento klasifikátor, ale bude dokonalým klasifikátorem, pokud zaměníme označení negativní třídy za pozitivní). Plocha AUROC náhodného klasifikátoru bude 0.5, neboť tento klasifikátor nedokáže vůbec rozlišovat mezi pozitivními a negativními případy.

6.9.3 Výsledky experimentu

Výsledné ROC křivky pro různé hodnoty SNR jsou zobrazeny v obrázcích 6.15-6.20. Modrá čára zobrazuje výsledky algoritmu ESE, zelená tečkovaná čára zobrazuje výsledky algoritmu LE, červená přerušovaná čára výsledky algoritmu ELBND a černá čerchovaná čára zobrazuje výsledky náhodného klasifikátoru.

Pro každou ROC křivku byla vypočtena hodnota plochy pod touto křivkou, AUROC, pomocí lichoběžníkové metody, jako

$$AUROC \approx \sum_{j=1}^{n_t} \frac{TPR(FPR(j)) + TPR(FPR(j+1))}{2} \cdot (FPR(j+1) - FPR(j)) \quad (6.30)$$

kde n_t reprezentuje počet vyhodnocovaných prahů zmenšený o 1. Výsledné plochy pod křivkami ROC jsou pro jednotlivé metody a směrodatné odchylky šumu uvedené v následující tabulce 6.1. Tučně je zvýrazněna největší hodnota AUROC. Podle uvedených hodnot při průměrném $SNR = 35.8$ nejlépe rozlišuje mezi pozitivními a negativními případy algoritmus LE. Pro nižší hodnoty SNR je nejlépe separujícím algoritmem ESE. V další tabulce 6.2 je uvedená úspěšnost klasifikace, kde za úspěšnou klasifikaci je považován případ, kdy maximální hodnota ESE, ELBND nebo LE během experimentu je v intervalu $200 \leq k \leq 210$, tedy do deseti vzorků po změně trendu. Tučně jsou zvýrazněny hodnoty nejvyšší úspěšnosti detekce. Z výsledků je patrné, že pro průměrné $SNR = 35.8$ má nejvyšší úspěšnost algoritmus LE.

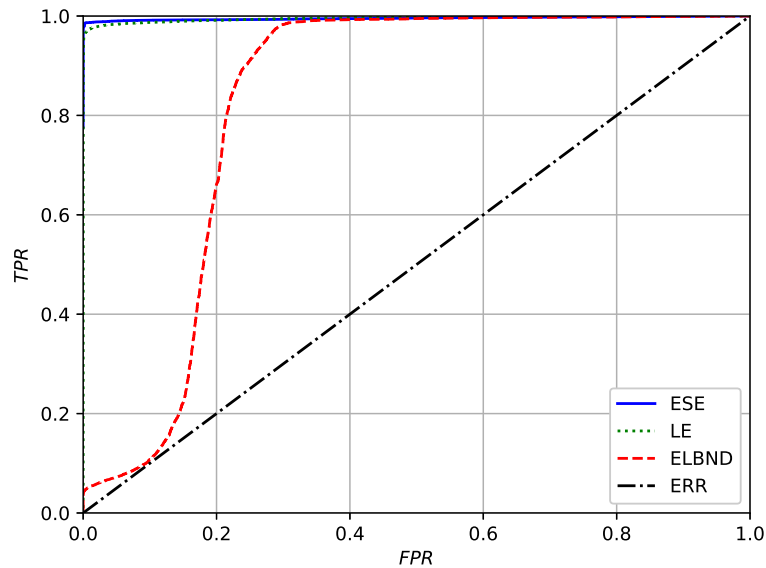
Tabulka 6.1: *AUROC* pro detekci změny trendu

		AUROC		
σ_n	SNR [dB]	<i>ESE</i>	<i>LE</i>	<i>ELBND</i>
0.1	35.8	0.9954	0.9952	0.8234
0.2	30.0	0.9920	0.9912	0.8299
0.5	21.7	0.9816	0.9777	0.8288
1.0	16.2	0.9576	0.9496	0.8263
2.0	10.8	0.9286	0.9214	0.8397
2.5	9.2	0.9134	0.9056	0.8446

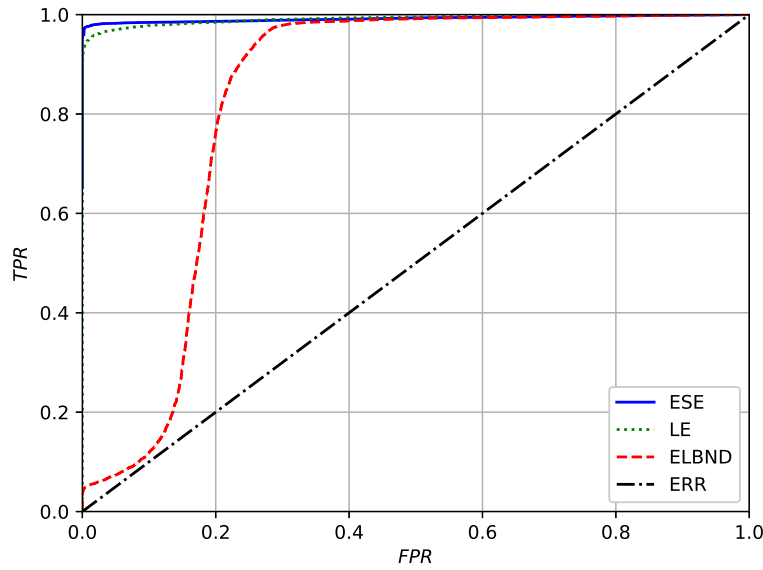
Tabulka 6.2: Úspěšnost detekce změny trendu

		Úspěšnost detekce		
σ_n	SNR [dB]	<i>ESE</i>	<i>LE</i>	<i>ELBND</i>
0.1	35.8	98.88	98.92	60.00
0.2	30.0	98.14	98.03	59.61
0.5	21.7	95.18	95.08	59.65
1.0	16.2	90.42	89.96	57.67
2.0	10.8	81.27	78.51	57.69
2.5	9.2	75.86	71.56	57.16

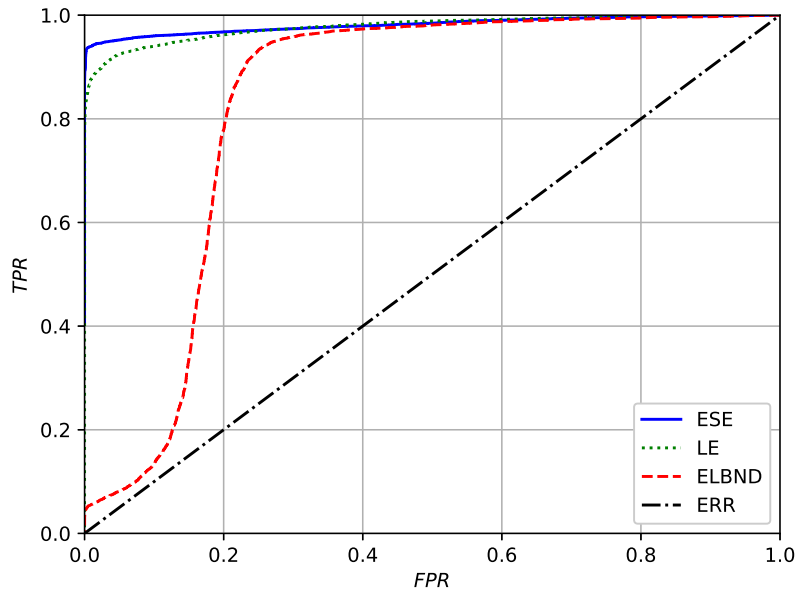
Pro nižší hodnoty SNR je algoritmem s nejvyšší úspěšností detekce algoritmus ESE.



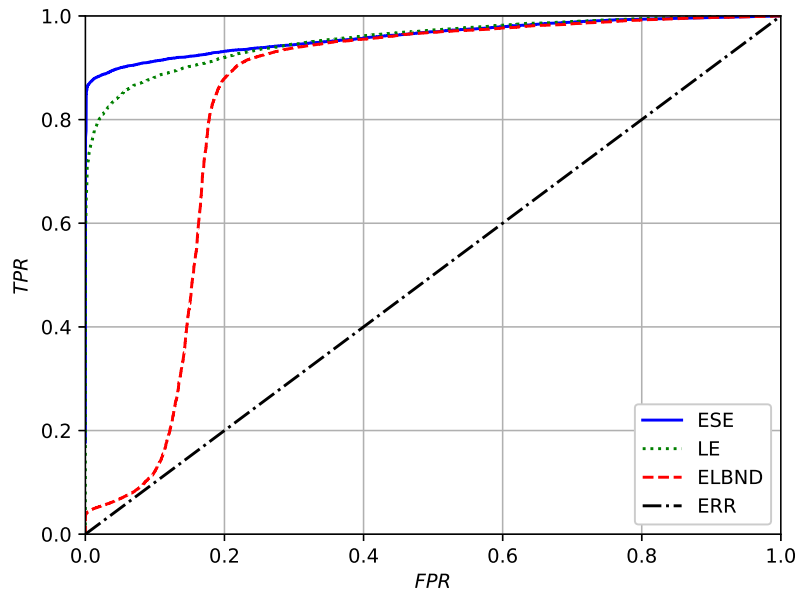
Obrázek 6.15: ROC křivky v případě detekce změny trendu signálu obsahujícího aditivní gaussianský šum se směrodatnou odchylkou $\sigma = 0.1$. Průměrná hodnota SNR experimentů byla $SNR = 35.80$ dB. Černá čerchovaná čára (RANDOM) reprezentuje náhodný klasifikátor.



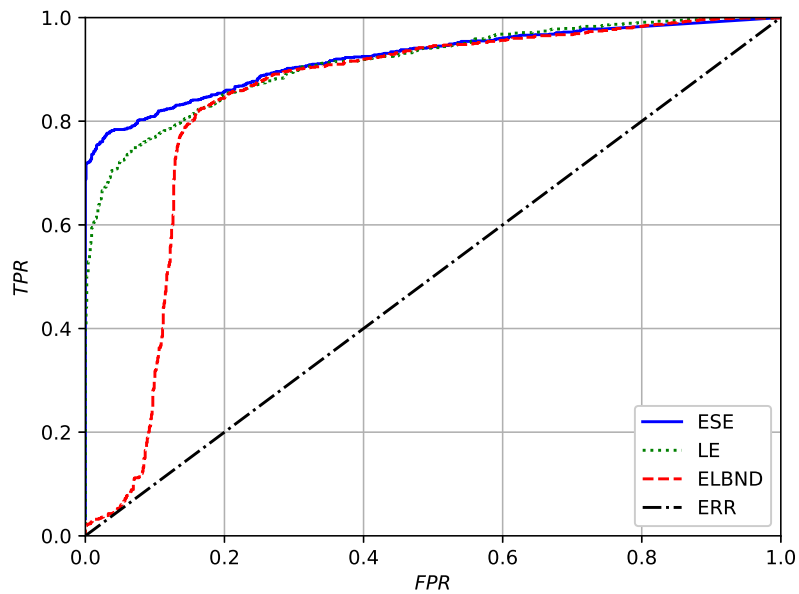
Obrázek 6.16: ROC křivky v případě detekce změny trendu signálu obsahujícího aditivní gaussianový šum se směrodatnou odchylkou $\sigma_n = .2$. Průměrná hodnota SNR experimentů byla $SNR = 30.00 \text{ dB}$. Černá čerchovaná čára (RANDOM) reprezentuje náhodný klasifikátor.



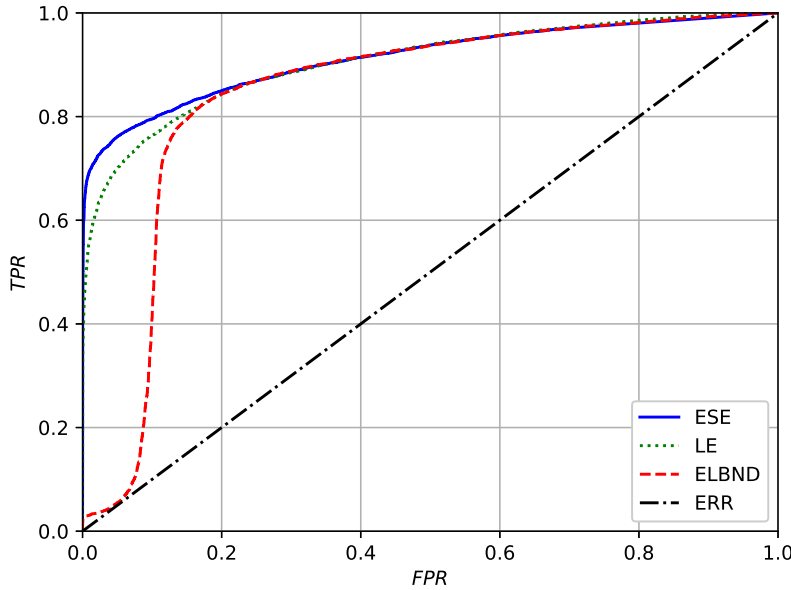
Obrázek 6.17: ROC křivky v případě detekce změny trendu signálu obsahujícího aditivní gaussianový šum se směrodatnou odchylkou $\sigma_n = 0.5$. Průměrná hodnota SNR experimentů byla $SNR = 21.70 \text{ dB}$. Černá čerchovaná čára (RANDOM) reprezentuje náhodný klasifikátor.



Obrázek 6.18: ROC křivky v případě detekce změny trendu signálu obsahujícího aditivní gaussovský šum se směrodatnou odchylkou $\sigma_n = 1.0$. Průměrná hodnota SNR experimentů byla $SNR = 16.20 \text{ dB}$. Černá čerchovaná čára (RANDOM) reprezentuje náhodný klasifikátor.



Obrázek 6.19: ROC křivky v případě detekce změny trendu signálu obsahujícího aditivní gaussovský šum se směrodatnou odchylkou $\sigma_n = 2.0$. Průměrná hodnota SNR experimentů byla $SNR = 10.88 \text{ dB}$. Černá čerchovaná čára (RANDOM) reprezentuje náhodný klasifikátor.



Obrázek 6.20: ROC křivky v případě detekce změny trendu signálu obsahujícího aditivní gaussovský šum se směrodatnou odchylkou $\sigma_n = 2.5$. Průměrná hodnota SNR experimentů byla $SNR = 9.20 \text{ dB}$. Černá čerchovaná čára (RANDOM) reprezentuje náhodný klasifikátor.

6.10 Vyhodnocení výpočetní náročnosti metod odhadu parametrů zobecněného Paretova rozdělení

Výsledky v této podkapitole byli publikovány v [V2]. Cílem bylo určit výpočetní čas výpočtu parametrů GPD v typické aplikaci pro použití algoritmu ESE, který byl, v tomto případě, testován ne experimentu detekce skokové změny parametrů generátoru signálu.

6.10.1 Motivace

Detekce novosti v reálném čase je úloha, která nalézá své uplatnění nejen v oblasti detekci a diagnostiky v průmyslových aplikacích [141], ale také např. v detekci narušení počítačových sítí [142] nebo v zabezpečovacích systémech [143]. Další oblastí uplatnění je např. mobilní robotika, která je specifická tím, že robot má k dispozici pouze limitovaný výpočetní výkon [144, 145]. Pro metody detekce novosti v reálném čase je tedy důležité, aby vynikali dostatečně nízkou výpočetní náročností. Z tohoto důvodu byli otestovány tři různé metody výpočtu parametrů GPD (viz kapitola 4, protože tento výpočet je z hlediska použití algoritmu ESE potenciálně limitující z hlediska využitelnosti v aplikacích detekce v reálném čase. Jmenovitě byli otestovány tyto metody: metoda maximální věrohodnosti (ML), metoda momentů (MOM) a metoda kvazi-maximální věrohodnosti (QML) (více viz kapitola 4). Výpočetní čas potřebný k určení parametrů GPD pomocí těchto metod byl vyhodnocen při experimentu, ve

kterém dojde ke skokové změně parametrů generátoru signálu.

6.10.2 Specifikace experimentu

Vzhledem k povaze experimentu, který slouží k vyhodnocení výpočetní náročnosti různých metod určení parametrů GPD, a nikoliv k detekci novosti v nějakém komplexním procesu, byl zvolen jednoduchý lineární kombinační filtr (LNU), jehož výstup v diskretním časovém okamžiku k je definován jako

$$\hat{y}(k) = w_1 \cdot x_1(k) + w_2 \cdot x_2(k) + w_3 \cdot x_3(k) \quad (6.31)$$

a tento filtr je adaptován algoritmem NLMS (viz kapitola 2.2.1), přičemž rychlost učení μ byla nastavena jako $\mu = 0.8$.

Pro výstup generátoru signálu platí vztah

$$y(k) = x_1(k) + x_2(k) + x_3(k) + v(k) \quad (6.32)$$

pro všechny $1 \leq k \leq 200$. Člen $v(k)$ reprezentuje aditivní gaussovský šum s nulovou střední hodnotou a směrodatnou odchylkou $\sigma_{noise} = 0.1$. V diskretním časovém okamžiku $k = 201$ dojde ke změně generátoru signálu a jeho výstup přejde do tvaru

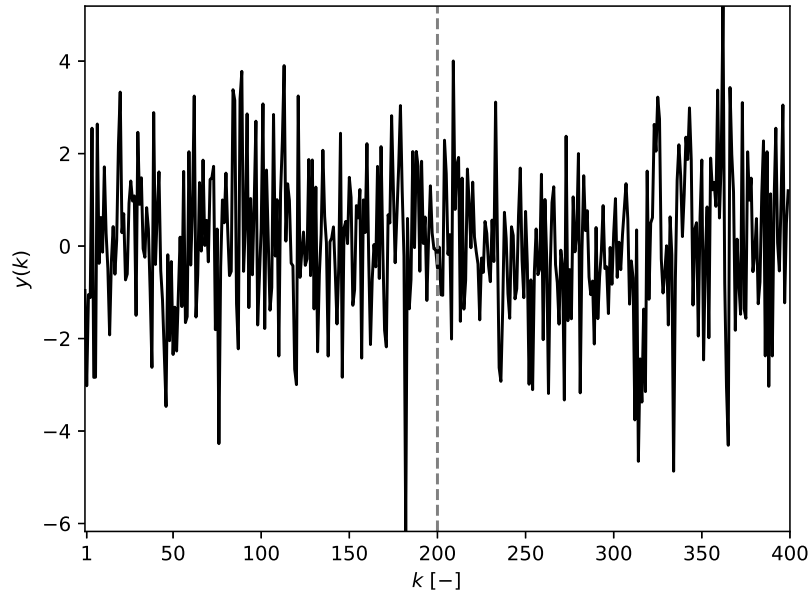
$$y(k) = 0.7 \cdot x_1(k) + 1.2 \cdot x_1(k) + 1.1 \cdot x_1(k) + v(k) \quad (6.33)$$

pro $201 \leq k \leq 400$. Hodnota všech vstupů generátoru signálu je v každém časovém okamžiku k vybrána ze standartního rozdělení normálního rozdělení, takže i -tý vstup $x_i \sim \mathcal{N}(0, 1)$. Změna parametrů signálu byla vybrána tak, aby nedošlo ke změně střední hodnoty signálu $y(k)$.

Délka okna pro odhad parametrů GPD byla během experimentu nastavena na $n_s = 1200$. Metoda POT byla zvolena podle rovnice 4.5. Před experimentem bylo pořízeno 1200 vzorků vygenerovaných generátorem signálu definovaným vztahem 6.31, na něž byla použita metoda POT, tak aby při experimentu v diskretním časovém okamžiku $k = 1$ byla hodnota ESE relevantní.

Průběh výstupní hodnoty filtru je zobrazen na obrázku 6.21, hodnota ESE potom na obrázku 6.22. Hodnoty parametrů ξ , σ , μ GPD, pro všechny tři adaptivní váhy, během experimentu jsou zobrazeny na obrázcích 6.24, 6.25 a 6.23.

Experiment byl proveden na PC s procesorem Intel(R) Core(TM) i5-7400 se 4mi jádry s taktovací frekvencí 3001 MHz a operační pamětí o velikosti 32 GB. Operační systém byl Windows 10 Pro, 64-bitová verze 10.0.18362. Kód byl napsán v Python 3.6.1 [146] a byly použity knihovny Numpy 1.17.0 [147] a Scipy 1.4.1 [148].



Obrázek 6.21: Výstup adaptivního filtru během experimentu. Skoková změna parametrů generátoru signálu je zvýrazněná svislou vodorovnou čarou v diskretním časovém okamžiku $k = 200$.

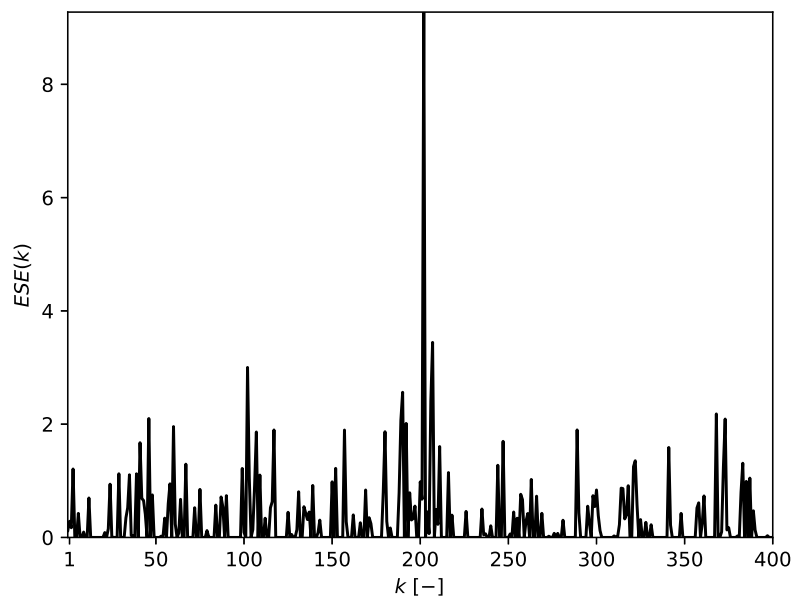
6.10.3 Výsledky a diskuze

Průměrný čas výpočtu \bar{t} parametrů všech tří GPD (počet GPD odpovídá počtu adaptivních parametrů filtru) a odpovídající směrodatné odchylky σ_t jsou uvedeny v následující tabulce 6.3. Čas výpočtu je určený pro jeden experiment (400 vzorků).

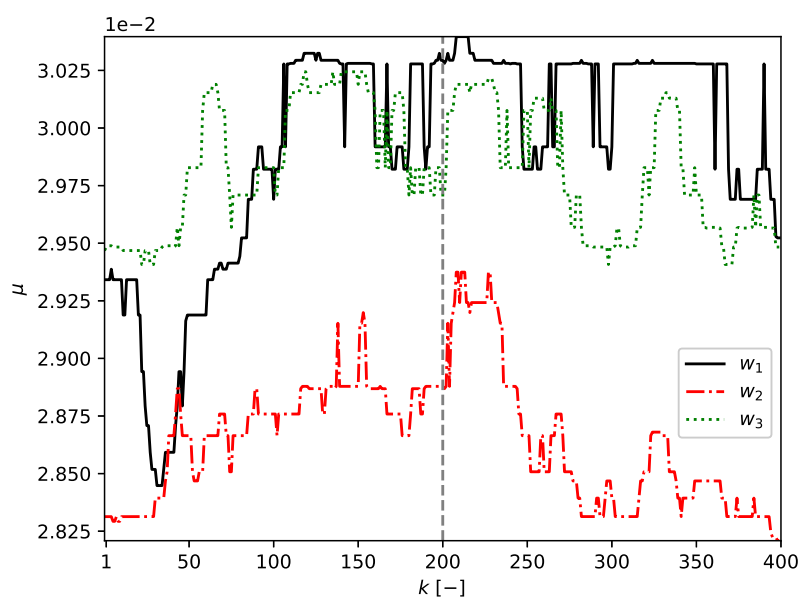
Tabulka 6.3: Tabulka průměrných časů výpočtu pro jednotlivé adaptivní váhy a odpovídajících směrodatných odchylek vybraných metod výpočtu parametrů GPD

	Metoda	\bar{t} [ms]	σ_t [ms]
w_1	ML	26.198	3.396
	QML	0.354	0.478
	MOM	0.076	0.264
w_2	ML	26.718	2.302
	QML	0.337	0.471
	MOM	0.064	0.244
w_3	ML	24.982	1.964
	QML	0.395	0.489
	MOM	0.060	0.238

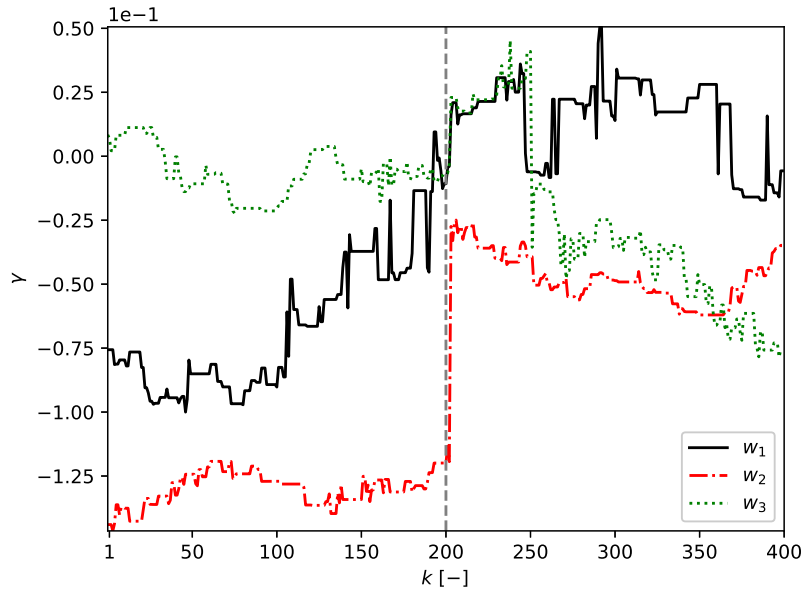
Z uvedených výsledků je patrné, že nejrychlejší metoda je MOM. Podstatnou nevýhodou této metody pro využití v aplikacích, které vyhodnocují data v reálném čase je její omezení na hodnoty parametrů GPD (viz kapitola 4). Pokud parametry uvedené omezení nesplňují,



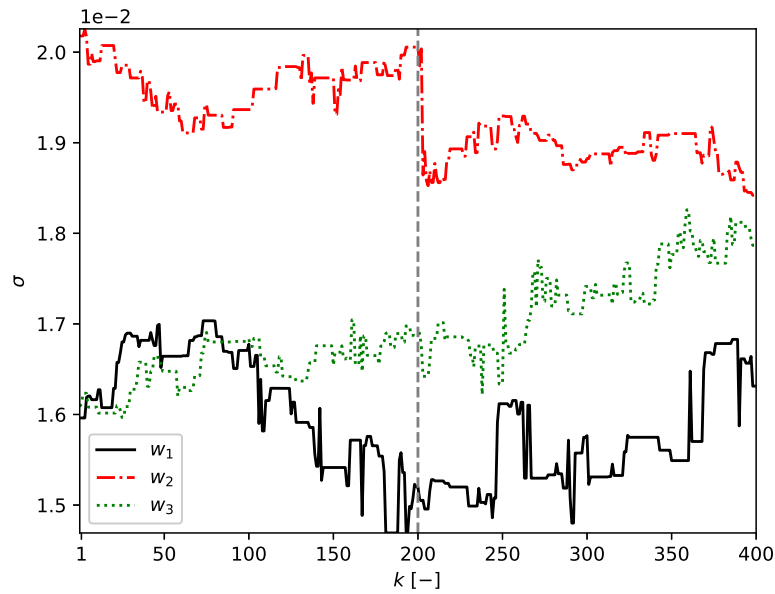
Obrázek 6.22: Hodnota ESE během experimentu. Globální maximum odpovídá změně parametrů generátoru signálu, resp. úspěšné detekci novosti.



Obrázek 6.23: Hodnota parametru μ GPD pro všechny tři adaptivní váhy w_1 , w_2 , w_3 během experimentu detekce změn parametrů generátoru signálu. Svislá čára v diskretním časovém okamžiku $k = 200$ znázorňuje skokovou změnu parametrů generátoru signálu.



Obrázek 6.24: Hodnota parametru γ GPD pro všechny tři adaptivní váhy w_1, w_2, w_3 během experimentu detekce změn parametrů generátoru signálu. Svislá čára v diskretním časovém okamžiku $k = 200$ znázorňuje skokovou změnu parametrů generátoru signálu.



Obrázek 6.25: Hodnota parametru σ GPD pro všechny tři adaptivní váhy w_1, w_2, w_3 během experimentu detekce změn parametrů generátoru signálu. Svislá čára v diskretním časovém okamžiku $k = 200$ znázorňuje skokovou změnu parametrů generátoru signálu.

vypočtené hodnoty nepřesné (resp. nesmyslné) a tedy nepoužitelné pro algoritmus ESE, který začne produkovat nepřesné výsledky. Z pohledu úlohy detekce novosti je diskutabilní, zda-li můžeme garantovat, že sledovaný proces po celou dobu bude splňovat uvedené omezení.

Metoda, jejíž výpočetní čas byl nejvyšší je ML, což je vzhledem k iterativnímu určení parametrů GPD očekávatelné. Nevýhoda použití této metody tkví v nemožnosti určit minimální resp. maximální počet iterací. Jednou z možností jak zrychlit nalezení parametrů je využití apriorní informace o hodnotách těchto parametrů. V rámci experimentu však byla využita pouze apriorní informace o parametru μ , který odpovídá nejmenší hodnotě přírůstku vah, které byli získány metodou POT aplikovanou na plovoucí okno délky n_s .

Dobrým kompromisem mezi výše uvedenými metodami je použití metody QML. Výpočetní čas této metody byl v uvedeném experimentu o dva řády kratší než ML a asi pětkrát delší než MOM. Pro každou z vyhodnocovaných vah byl kratší než $500 \mu s$.

Z provedeného experimentu je patrné, že použití algoritmu ESE pro aplikace v reálném čase je limitováno počtem parametrů filtru a rychlostí vzorkování monitorovaného procesu.

7 Závěr

Předložená dizertační práce je věnována použití adaptivních systémů při analýze dat. Z chronologického pohledu je prvním výstupem během zpracování této práce případová studie algoritmu Learning Entropy pro detekci změn stavu bioprocesu s využitím adaptivního fuzzy filtru [V4], více viz příloha B. Dále byla úspěšně vyzkoušena detekce fázové změny v krystalografických datech pomocí adaptivního filtru a zobecněného rozdělení extrémních hodnot [V5]. Zobecněné rozdělení extrémních hodnot bylo také využito v případové studii skokové změny parametrů generátoru signálu [V6].

Za zásadní výsledek lze považovat nový originální algoritmus pro detekci novosti, který vyhodnocuje přírůstky adaptivních vah filtru, Extreme Seeking Entropy [V1] (viz kapitola 5). Tento algoritmus byl otestován v následujících případových studiích (viz kapitola 6): detekce pertubace v chaotické časové řadě získané řešením Mackey-Glassovy rovnice, detekce změny rozptylu šumu v náhodném datovém toku, detekce skokové změny parametrů generátoru signálu, detekce náhlé absence šumu, detekce změny trendu a při detekci epilepsie v myším EEG. Pro detekci skokové změny trendu a skokové změny parametrů signálu byla vyhodnocena úspěšnost této detekce a výsledky porovnány s výsledky algoritmů Learning Entropy a Error and Learning Based Novelty Detection, přičemž v obou případech byla úspěšnost detekce algoritmu ESE vyšší pro téměř všechny vyhodnocované hodnoty SNR. Pro hodnotu $SNR > 34 \text{ dB}$ dosáhl algoritmus při detekci skokové změny parametrů generátoru signálu 100% úspěšnost. Při detekci změny trendu měl algoritmus ESE pro hodnoty $SNR > 8 \text{ dB}$ větší úspěšnost detekce než srovnávané algoritmy LE a ELBND. Výše uvedené výsledky byly publikovány v [V1].

Pro možné použití v aplikacích detekce v reálném čase byla experimentálně zjišťována výpočetní časová náročnost různých metod odhadů parametrů zobecněného Paretova rozdělení v případě použití algoritmu ESE při detekci skokové změny parametrů [V2]. Výsledkem je porovnání 3 různých metod odhadu parametrů. Limitujícím faktorem použití ESE v reálném čase je v zásadě počet adaptivních parametrů filtru, které je potřeba vyhodnocovat a samozřejmě rychlost vzorkování monitorovaného signálu.

Pro odhad úspěšnosti detekce novosti pomocí algoritmu ESE byla také vyhodnocena ROC křivka v případě detekce změny trendu signálu s různými poměry SNR a byly určeny příslušné

plochy pod těmito ROC křivkami [V3]. Dosažené výsledky byly opět porovnány s algoritmy LE a ELBND a bylo ověřeno, že pro hodnoty $SNR \leq 30 \text{ dB}$ dosahuje algoritmus ESE lepších výsledků. Pro vyšší hodnoty SNR pak byly výsledky ESE srovnatelné s výsledky LE. Cílem této studie bylo zjistit jak dobře dokáže algoritmus ESE separovat nová data v závislosti na volbě prahu, který rozhoduje o tom zda data obsahují novost či nikoliv. Pro představu ještě uveďme, že např. pro hodnotu $SNR = 16.2 \text{ dB}$ bylo dosaženo úspěšnosti detekce 90.42%.

Stanovené cíle dizertační práce (viz kapitola 1.1) tak lze, na základě výše uvedených výsledků, považovat za splněné.

7.1 Možné směry budoucího výzkumu

V budoucnu se nabízí rozvíjet téma využití adaptivních systémů ve zpracování dat několika směry. Potenciální využití vyhodnocení změn vah adaptivních systémů lze využít v optimalizaci velikosti datasetů v oblasti hlubokého učení, což by mohlo výrazně snížit časovou náročnost učení hlubokých sítí. Za účelem snížení výpočetního času algoritmu ESE je potřeba vyzkoušet další metody odhadu parametrů zobecněného Paretova rozdělení a vyzkoušet adaptivní metody volby velikosti prahu pro metodu Peak-over-threshold. Zajímavým tématem je také vliv šumu a jeho typu na velikost přírůstků adaptivních vah filtrů a jejich pravděpodobnostní rozdělení. V neposlední řadě se nabízí otázka, jak ovlivní typ kritériální funkce pro optimalizaci adaptivního filtru výsledky algoritmu ESE a je-li možné různé kritériální funkce využívat k detekci novosti, případně pomocí nich typ novosti klasifikovat.

Publikace autora

- [V1] VRBA, Jan; MAREŠ, Jan. Introduction to Extreme Seeking Entropy. *Entropy*, 2020, 22.1: 93.
- [V2] VRBA, Jan; MAREŠ, Jan. Computational Performance of the Parameters Estimation in Extreme Seeking Entropy Algorithm. In: *2020 International Conference on Applied Electronics (AE)*. IEEE, 2020. p. 1-4.
- [V3] VRBA, Jan; MAREŠ, Jan. ROC Analysis of Extreme Seeking Entropy for Trend Change Detection. In: *2020 International Conference on Applied Electronics (AE)*. IEEE, 2020. p. 1-4.
- [V4] VRBA, Jan. Využití fuzzy systémů a algoritmu learning entropy pro detekci změn stavů bioprosesu. In: *Automatizacia a riadenie v teorii a praxi ARTEP 2017*. Technická univerzita Košice, 2017.
- [V5] VRBA, Jan. *XLIII. Seminář ASŘ - Adaptivní metoda detekce* [přednáška]. Ostrava: VŠB TU Ostrava, 27.4.2018.
- [V6] VRBA, Jan. Adaptive Novelty Detection with Generalized Extreme Value Distribution. In: *2018 International Conference on Applied Electronics (AE)*. IEEE, 2018. p. 1-4.
- [V7] BUKOVSKÝ, Ivo, et al. Study of learning entropy for onset detection of epileptic seizures in EEG time series. In: *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2016. p. 3302-3305.
- [V8] OSWALD, Cyril, et al. Novelty Detection in System Monitoring and Control with HONU. In: *Applied Artificial Higher Order Neural Networks for Control and Recognition*. IGI Global, 2016. p. 61-78.
- [V9] VRBA, Jan, et al. An Automated Platform for Microrobot Manipulation. In: *International Workshop on Soft Computing Models in Industrial and Environmental Applications*. Springer, Cham, 2020. p. 255-265.

- [V10] BÍLA, Jiří; VRBA, Jan. The Detection and Interpretation of Emergent Situations in ECG Signals. In: *International Conference on Soft Computing-MENDEL*. Springer, Cham, 2016. p. 264-275.
- [V11] MOJZES, Matej, et al. Feature selection via competitive levy flights. In: *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2016. p. 3731-3736.
- [V12] BÍLA, Jiří; NOVÁK, Martin; VRBA, Jan. Detection of emergent situations in complex systems represented by algebras of transformations. In: *MATEC Web of Conferences*. EDP Sciences, 2016. p. 02035.
- [V13] BUKOVSKÝ, Ivo, OSWALD, Cyril, VRBA, Jan. Případová studie použití entropie učení pro adaptivní detekci při řízení spalování tuhých paliv. In: *Automatizácia a riadenie v teórii a praxi 2015*. Technická univerzita Košice, 2015.

Literatura

- [1] SHARMA, Anish and ANDREWS, Rebecca. Managing-Exponential-Data-Growth-and-Application-Modernization. *IBM* [online]. 11 November 1999. [cit. 11.8.2020]. Dostupné z: <https://www.ibm.com/cloud/blog/managing-exponential-data-growth-and-application-modernization>
- [2] HAYKIN, Simon S. *Adaptive filter theory*. Pearson Education India, 2008.
- [3] PROAKIS, John G. a Dimitris G. MANOLAKIS. *Digital signal processing: principles, algorithms, and applications*. Pearson Education India, 2001.
- [4] STRICHARTZ, Robert S. *A guide to distribution theory and Fourier transforms*. World Scientific Publishing Company, 2003.
- [5] GERSHO, Allen; GRAY, Robert M. *Vector quantization and signal compression*. Springer Science & Business Media, 2012.
- [6] OOSTENVELD, Robert; PRAAMSTRA, Peter. The five percent electrode system for high-resolution EEG and ERP measurements. *Clinical neurophysiology*, 2001, 112.4: 713-719.
- [7] GUPTA, Madan M., et al. Fundamentals of higher order neural networks for modeling and simulation. In: *Artificial Higher Order Neural Networks for Modeling and Simulation*. IGI Global, 2013. p. 103-133.
- [8] WANG, L.-X.; MENDEL, Jerry M. Fuzzy adaptive filters, with application to nonlinear channel equalization. *IEEE Transactions on Fuzzy Systems*, 1993, 1.3: 161-170.
- [9] WANG, L.-X. *A course in fuzzy systems and control*. Upper Saddle River : Prentice-Hall, 1997.
- [10] MANDIC, Danilo P.; KANNA, Sithan; CONSTANTINIDES, Anthony G. On the intrinsic relationship between the least mean square and Kalman filters [Lecture Notes]. *IEEE Signal Processing Magazine*, 2015, 32.6: 117-122.
- [11] ROWELL, D. 2.161 Signal Processing: Continuous and Discrete, Fall 2008. 2008.

- [12] MILJKOVIĆ, Dubravko. Review of novelty detection methods. In: *The 33rd International Convention MIPRO*. IEEE, 2010. p. 593-598.
- [13] ZIMEK, Arthur; FILZMOSER, Peter. There and back again: Outlier detection between statistical reasoning and data mining algorithms. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2018, 8.6: e1280.
- [14] MARKOU, Markos; SINGH, Sameer. Novelty detection: a review—part 1: statistical approaches. *Signal processing*, 2003, 83.12: 2481-2497.
- [15] MARKOU, Markos; SINGH, Sameer. Novelty detection: a review—part 2:: neural network based approaches. *Signal processing*, 2003, 83.12: 2499-2521.
- [16] PIMENTEL, Marco AF, et al. A review of novelty detection. *Signal Processing*, 2014, 99: 215-249.
- [17] REYNOLDS, Douglas A. Gaussian Mixture Models. *Encyclopedia of biometrics*, 2009, 741.
- [18] HEYNS, Theo; HEYNS, Philippus Stephanus; DE VILLIERS, Johan Pieter. Combining synchronous averaging with a Gaussian mixture model novelty detection scheme for vibration-based condition monitoring of a gearbox. *Mechanical Systems and Signal Processing*, 2012, 32: 200-215.
- [19] BISHOP, Christopher M. *Pattern recognition and machine learning*. Springer, 2006.
- [20] DUDA, Richard O.; HART, Peter E.; STORK, David G. *Pattern classification*. John Wiley & Sons, 2012.
- [21] DOAN, AnHai; HALEVY, Alon; IVES, Zachary. *Principles of data integration*. Elsevier, 2012.
- [22] ZHANG, Yu, et al. Detection of emerging faults on industrial gas turbines using extended Gaussian mixture models. *International Journal of Rotating Machinery*, 2017, 2017.
- [23] NASIOS, Nikolaos; BORS, Adrian G. Variational learning for Gaussian mixture models. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 2006, 36.4: 849-862.
- [24] MIYAZAKI, Bungo, et al. Change detection of orders in stock markets using a Gaussian mixture model. *Intelligent Systems in Accounting, Finance and Management*, 2014, 21.3: 169-191.

- [25] DEMPSTER, Arthur P.; LAIRD, Nan M.; RUBIN, Donald B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 1977, 39.1: 1-22.
- [26] WATANABE, Hidenori; MURAMATSU, Shogo; KIKUCHI, Hisakazu. Interval calculation of EM algorithm for GMM parameter estimation. In: *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*. IEEE, 2010. p. 2686-2689.
- [27] BOUGUILA, Nizar a Wentao FAN, ed. *Mixture Models and Applications* Springer International Publishing, 2020.
- [28] ROBERTS, Stephen J. Novelty detection using extreme value statistics. *IEE Proceedings-Vision, Image and Signal Processing*, 1999, 146.3: 124-129.
- [29] FISHER, Ronald Aylmer; TIPPETT, Leonard Henry Caleb. Limiting forms of the frequency distribution of the largest or smallest member of a sample. In: *Mathematical Proceedings of the Cambridge Philosophical Society*. Cambridge University Press, 1928. p. 180-190.
- [30] PICKANDS III, James, et al. Statistical inference using extreme order statistics. *The Annals of Statistics*, 1975, 3.1: 119-131.
- [31] BALKEMA, August A.; DE HAAN, Laurens. Residual life time at great age. *The Annals of probability*, 1974, 792-804.
- [32] CLIFTON, David Andrew; HUGUENY, Samuel; TARASSENKO, Lionel. Novelty detection with multivariate extreme value statistics. *Journal of signal processing systems*, 2011, 65.3: 371-389.
- [33] MAHALANOBIS, Prasanta Chandra. *On the generalized distance in statistics*. National Institute of Science of India, 1936.
- [34] CLIFTON, David A., et al. Extending the generalised pareto distribution for novelty detection in high-dimensional spaces.
- [35] GREWAL, Mohinder S.; ANDREWS, Angus P. *Kalman filtering: Theory and Practice with MATLAB*. John Wiley & Sons, 2014.
- [36] LEE, Hyoung-joo; ROBERTS, Stephen J. On-line novelty detection using the kalman filter and extreme value theory. In: *2008 19th International Conference on Pattern Recognition*. IEEE, 2008. p. 1-4. *Journal of signal processing systems*, 2014, 74.3: 323-339.
- [37] SIFFER, Alban, et al. Anomaly detection in streams with extreme value theory. In: *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2017. p. 1067-1075.

- [38] HUANG, Yan, et al. Generalized Pareto Model Based on Particle Swarm Optimization for Anomaly Detection. *IEEE Access*, 2019, 7: 176329-176338.
- [39] KENNEDY, James; EBERHART, Russell. Particle swarm optimization. In: *Proceedings of ICNN'95-International Conference on Neural Networks*. IEEE, 1995. p. 1942-1948.
- [40] LUCA, Stijn; CLIFTON, David A.; VANRUMSTE, Bart. One-class classification of point patterns of extremes. *The Journal of Machine Learning Research*, 2016, 17.1: 6581-6601.
- [41] GALES, Mark; YOUNG, Steve. *The application of hidden Markov models in speech recognition*. Now Publishers Inc, 2008.
- [42] FINK, Gernot A. *Markov models for pattern recognition: from theory to applications*. Springer Science & Business Media, 2014.
- [43] KOSKI, Timo. *Hidden Markov models for bioinformatics*. Springer Science & Business Media, 2001.
- [44] MACDONALD, Iain L.; ZUCCHINI, Walter. *Hidden Markov and other models for discrete-valued time series*. CRC Press, 1997.
- [45] BHAR, Ramaprasad; HAMORI, Shigeyuki. *Hidden Markov models: applications to financial economics*. Springer Science & Business Media, 2004.
- [46] WARRENDER, Christina; FORREST, Stephanie; PEARLMUTTER, Barak. Detecting intrusions using system calls: Alternative data models. In: *Proceedings of the 1999 IEEE symposium on security and privacy (Cat. No. 99CB36344)*. IEEE, 1999. p. 133-145.
- [47] ZOHREVAND, Zahra, et al. Hidden Markov based anomaly detection for water supply systems. In: *2016 IEEE International Conference on Big Data (Big Data)*. IEEE, 2016. p. 1551-1560.
- [48] YU, Jianbo. Adaptive hidden Markov model-based online learning framework for bearing faulty detection and performance degradation monitoring. *Mechanical Systems and Signal Processing*, 2017, 83: 149-162.
- [49] SCHMIDT, Stephan; HEYNS, Philippus Stephanus; DE VILLIERS, Johan Pieter. A novelty detection diagnostic methodology for gearboxes operating under fluctuating operating conditions using probabilistic techniques. *Mechanical Systems and Signal Processing*, 2018, 100: 152-166.
- [50] ABDI, Hervé; WILLIAMS, Lynne J. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2010, 2.4: 433-459.

- [51] LI, Jinbo; PEDRYCZ, Witold; JAMAL, Iqbal. Multivariate time series anomaly detection: A framework of Hidden Markov Models. *Applied Soft Computing*, 2017, 60: 229-240.
- [52] TOLOUE, Kamran Fartash; JAHAN, Majid Vafaei. Anomalous behavior detection of marine vessels based on Hidden Markov Model. In: *2018 6th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS)*. IEEE, 2018. p. 10-12.
- [53] ZHAO, Jingling, et al. Software Abnormal Behavior Detection Based on Hidden Markov Model. In: *International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*. Springer, Cham, 2017. p. 929-940.
- [54] LI, Peng, et al. Application of hidden Markov model in SQL injection detection. In: *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*. IEEE, 2017. p. 578-583.
- [55] ROBINSON, William N.; ARIA, Andrea. Sequential fraud detection for prepaid cards using hidden Markov model divergence. *Expert Systems with Applications*, 2018, 91: 235-251.
- [56] CORTES, Corinna; VAPNIK, Vladimir. Support-vector networks. *Machine learning*, 1995, 20.3: 273-297.
- [57] BOSER, Bernhard E.; GUYON, Isabelle M.; VAPNIK, Vladimir N. A training algorithm for optimal margin classifiers. In: *Proceedings of the fifth annual workshop on Computational learning theory*. 1992. p. 144-152.
- [58] GUPTA, Madan; JIN, Liang; HOMMA, Noriyasu. *Static and dynamic neural networks: from fundamentals to advanced theory*. John Wiley & Sons, 2004.
- [59] MOHRI, Mehryar; ROSTAMIZADEH, Afshin; TALWALKAR, Ameet. *Foundations of machine learning*. MIT press, 2018.
- [60] PEDREGOSA, Fabian, et al. Scikit-learn: Machine learning in Python. *The Journal of machine Learning research*, 2011, 12: 2825-2830.
- [61] MANNING, Christopher D.; SCHÜTZE, Hinrich; RAGHAVAN, Prabhakar. *Introduction to information retrieval*. Cambridge university press, 2008.
- [62] CRISTIANINI, Nello, et al. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.
- [63] AISERMAN, M. A.; BRAVERMAN, E. M.; ROZONOER, L. I. Theoretical foundations of the potential function method in pattern recognition. *Avtomat. i Telemekh*, 1964, 25.6: 917-936.

- [64] SCHÖLKOPF, Bernhard, et al. Support vector method for novelty detection. In: *Advances in neural information processing systems*. 2000. p. 582-588.
- [65] YAN, Ke; JI, Zhiwei; SHEN, Wen. Online fault detection methods for chillers combining extended kalman filter and recursive one-class SVM. *Neurocomputing*, 2017, 228: 205-212.
- [66] SADOOGHI, Mohammad Saleh; KHADEM, Siamak Esmailzadeh. Improving one class support vector machine novelty detection scheme using nonlinear features. *Pattern Recognition*, 2018, 83: 14-33.
- [67] PINCUS, Steven M. Approximate entropy as a measure of system complexity. *Proceedings of the National Academy of Sciences*, 1991, 88.6: 2297-2301.
- [68] RICHMAN, Joshua S.; MOORMAN, J. Randall. Physiological time-series analysis using approximate entropy and sample entropy. *American Journal of Physiology-Heart and Circulatory Physiology*, 2000, 278.6: H2039-H2049.
- [69] BANDT, Christoph; POMPE, Bernd. Permutation entropy: a natural complexity measure for time series. *Physical review letters*, 2002, 88.17: 174102.
- [70] SOLARZ, A., et al. Automated novelty detection in the WISE survey with one-class support vector machines. *Astronomy & Astrophysics*, 2017, 606: A39.
- [71] KOHONEN, Teuvo. The self-organizing map. *Proceedings of the IEEE*, 1990, 78.9: 1464-1480.
- [72] AGUAYO, Leonardo; BARRETO, Guilherme A. Novelty detection in time series using self-organizing neural networks: a comprehensive evaluation. *Neural Processing Letters*, 2018, 47.2: 717-744.
- [73] YIN, Hujun. Nonlinear dimensionality reduction and data visualization: a review. *International Journal of Automation and Computing*, 2007, 4.3: 294-303.
- [74] LI, Menglu; KASHEF, Rasha; IBRAHIM, Ahmed. Multi-Level Clustering-Based Outlier's Detection (MCOD) Using Self-Organizing Maps. *Big Data and Cognitive Computing*, 2020, 4.4: 24.
- [75] GAN, Guojun; MA, Chaoqun; WU, Jianhong. Data clustering: theory, algorithms, and applications. *Society for Industrial and Applied Mathematics*, 2007.
- [76] OLSZEWSKI, Dominik. Fraud detection using self-organizing map visualizing the user profiles. *Knowledge-Based Systems*, 2014, 70: 324-334.

- [77] BERNARD, Yann; HUEBER, Nicolas; GIRAU, Bernard. Novelty detection with self-organizing maps for autonomous extraction of salient tracking features. In: *International Workshop on Self-Organizing Maps*. Springer, Cham, 2019. p. 100-109.
- [78] TAYLOR, John G. Neural ‘bubble’ dynamics in two dimensions: foundations. *Biological cybernetics*, 1999, 80.6: 393-409.
- [79] CHI, Dongxiang. Self-organizing map-based color image segmentation with k-means clustering and saliency map. *International Scholarly Research Notices*, 2011, 2011.
- [80] RATRE, Avinash; PANKAJAKSHAN, Vinod. Tucker visual search-based hybrid tracking model and fractional Kohonen self-organizing map for anomaly localization and detection in surveillance videos. *The Imaging Science Journal*, 2018, 66.4: 195-210.
- [81] RIGAMONTI, Marco, et al. Identification of the degradation state for condition-based maintenance of insulated gate bipolar transistors: A self-organizing map approach. *Microelectronics Reliability*, 2016, 60: 48-61.
- [82] NAM, Tran Manh, et al. Self-organizing map-based approaches in DDoS flooding detection using SDN. In: *2018 International Conference on Information Networking (ICOIN)*. IEEE, 2018. p. 249-254.
- [83] ELMOUGY, Samir, et al. A parameter based growing ensemble of self-organizing maps for outlier detection in healthcare. *Cluster Computing*, 2019, 22.1: 2437-2460.
- [84] MATIĆ, Frano; KALINIĆ, Hrvoje; VILIBIĆ, Ivica. Interpreting Self-Organizing Map errors in the classification of ocean patterns. *Computers & geosciences*, 2018, 119: 9-17.
- [85] SIMONYAN, Karen; ZISSERMAN, Andrew. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [86] GOODFELLOW, Ian, et al. *Deep learning*. Cambridge: MIT press, 2016.
- [87] TANG, Zhiyi, et al. Convolutional neural network-based data anomaly detection method using multiple information for structural health monitoring. *Structural Control and Health Monitoring*, 2019, 26.1: e2296.
- [88] STAAR, Benjamin; LÜTJEN, Michael; FREITAG, Michael. Anomaly detection with convolutional neural networks for industrial surface inspection. *Procedia CIRP*, 2019, 79: 484-489.
- [89] BOZKURT, Baris; GERMANAKIS, Ioannis; STYLIANOU, Yannis. A study of time-frequency features for CNN-based automatic heart sound classification for pathology detection. *Computers in biology and medicine*, 2018, 100: 132-143.

- [90] SABOKROU, Mohammad, et al. Adversarially learned one-class classifier for novelty detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018. p. 3379-3388.
- [91] YAHAYA, Salisu Wada; LOTFI, Ahmad; MAHMUD, Mufti. A consensus novelty detection ensemble approach for anomaly detection in activities of daily living. *Applied Soft Computing*, 2019, 83: 105613.
- [92] GHOSAL, Tirthankar, et al. Novelty goes deep. A deep neural solution to document level novelty detection. In: *Proceedings of the 27th International Conference on Computational Linguistics*. 2018. p. 2802-2813.
- [93] OZA, Poojan; PATEL, Vishal M. One-class convolutional neural network. *IEEE Signal Processing Letters*, 2018, 26.2: 277-281.
- [94] SCHULTHEISS, Alexander, et al. Finding the unknown: Novelty detection with extreme value signatures of deep neural activations. In: *German Conference on Pattern Recognition*. Springer, Cham, 2017. p. 226-238.
- [95] NICOLAU, Miguel, et al. A hybrid autoencoder and density estimation model for anomaly detection. In: *International Conference on Parallel Problem Solving from Nature*. Springer, Cham, 2016. p. 717-726.
- [96] AN, Jinwon; CHO, Sungzoon. Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE*, 2015, 2.1: 1-18.
- [97] MARCHI, Erik, Fabio VESPERINI, Florian EYBEN, Stefano SQUARTINI a Bjorn SCHULLER. A novel approach for automatic acoustic novelty detection using a denoising autoencoder with bidirectional LSTM neural networks. In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* 2015, p. 1996-2000.
- [98] FENG, Qingsong, et al. Anomaly detection of spectrum in wireless communication via deep auto-encoders. *The Journal of Supercomputing*, 2017, 73.7: 3161-3178.
- [99] DOMINGUES, Rémi, et al. Deep Gaussian Process autoencoders for novelty detection. *Machine Learning*, 2018, 107.8-10: 1363-1383.
- [100] BHATTACHARJEE, Supritam; MANDAL, Devraj; BISWAS, Soma. Autoencoder based novelty detection for generalized zero shot learning. In: *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2019. p. 3646-3650.
- [101] KIEU, Tung, et al. Outlier Detection for Time Series with Recurrent Autoencoder Ensembles. In: *IJCAI*. 2019. p. 2725-2732.

- [102] MARCHI, Erik, et al. Deep recurrent neural network-based autoencoders for acoustic novelty detection. *Computational intelligence and neuroscience*, 2017, 2017.
- [103] KAWACHI, Yuta; KOIZUMI, Yuma; HARADA, Noboru. Complementary set variational autoencoder for supervised anomaly detection. In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018. p. 2366-2370.
- [104] WANG, Xuhong, et al. adVAE: A self-adversarial variational autoencoder with Gaussian anomaly prior knowledge for anomaly detection. *Knowledge-Based Systems*, 2020, 190: 105187.
- [105] DINIZ, Paulo Sergio Ramirez. *Adaptive filtering: algorithms and practical implementation*. Springer, 2008.
- [106] GOLLAMUDI, Sridhar, et al. Set-membership filtering and a set-membership normalized LMS algorithm with an adaptive step size. *IEEE Signal Processing Letters*, 1998, 5.5: 111-114.
- [107] DINIZ, Paulo SR; YAZDANPANA, Hamed. Data censoring with set-membership algorithms. In: *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2017. p. 121-125.
- [108] ZHOU, Bo, et al. Multi-sensor fusion robust localization for indoor mobile robots based on a set-membership estimator. In: *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*. IEEE, 2015. p. 157-162.
- [109] JAULIN, Luc. Robust set-membership state estimation; application to underwater robotics. *Automatica*, 2009, 45.1: 202-206.
- [110] HUANG, Jian; WANG, Yuan; FUKUDA, Toshio. Set-membership-based fault detection and isolation for robotic assembly of electrical connectors. *IEEE Transactions on Automation Science and Engineering*, 2016, 15.1: 160-171.
- [111] BUKOVSKY, Ivo. Learning entropy: Multiscale measure for incremental learning. *Entropy*, 2013, 15.10: 4159-4187.
- [112] BUKOVSKY, Ivo; KINSNER, Witold; HOMMA, Noriyasu. Learning Entropy as a Learning-Based Information Concept. *Entropy*, 2019, 21.2: 166.
- [113] CEJNEK, Matous; BUKOVSKY, Ivo. Concept drift robust adaptive novelty detection for data streams. *Neurocomputing*, 2018, 309: 46-53.
- [114] CEJNEK, Matous; BUKOVSKY, Ivo. 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. p. 373-377.

- [115] CEJNEK, Matous; BUKOVSKY, Ivo; VYSATA, Oldrich. Adaptive classification of EEG for dementia diagnosis. In: *2015 International Workshop on Computational Intelligence for Multimedia Understanding (IWCIM)*. IEEE, 2015. p. 1-5.
- [116] GRUBBS, Frank E. Procedures for detecting outlying observations in samples. *Technometrics*, 1969, 11.1: 1-21.
- [117] LIMPET, Eckhard; STAHEL, Werner A. Problems with using the normal distribution—and ways to improve quality and efficiency of data analysis. *PLoS One*, 2011, 6.7: e21403.
- [118] FORBES, Catherine, et al. *Statistical distributions*. John Wiley & Sons, 2011.
- [119] COLES, Stuart, et al. *An introduction to statistical modeling of extreme values*. London: Springer, 2001.
- [120] LEE, Hyoung-joo; ROBERTS, Stephen J. On-line novelty detection using the kalman filter and extreme value theory. In: *2008 19th International Conference on Pattern Recognition*. IEEE, 2008. p. 1-4.
- [121] SCARROTT, Carl; MACDONALD, Anna. A review of extreme value threshold estimation and uncertainty quantification. *REVSTAT—Statistical Journal*, 2012, 10.1: 33-60.
- [122] FERREIRA, A.; DE HAAN, Laurens; PENG, Liang. On optimising the estimation of high quantiles of a probability distribution. *Statistics*, 2003, 37.5: 401-434.
- [123] LORETAN, Mico; PHILLIPS, Peter CB. Testing the covariance stationarity of heavy-tailed time series: An overview of the theory with applications to several financial datasets. *Journal of empirical finance*, 1994, 1.2: 211-248.
- [124] HOSKING, Jonathan RM; WALLIS, James R. Parameter and quantile estimation for the generalized Pareto distribution. *Technometrics*, 1987, 29.3: 339-349.
- [125] DUMOUCHEL, William H. Estimating the stable index α in order to measure tail thickness: a critique. *The Annals of Statistics*, 1983, 1019-1031.
- [126] LUCENO, Alberto. Fitting the generalized Pareto distribution to data using maximum goodness-of-fit estimators. *Computational Statistics & Data Analysis*, 2006, 51.2: 904-917.
- [127] ZHANG, Jin. Likelihood moment estimation for the generalized Pareto distribution. *Australian & New Zealand Journal of Statistics*, 2007, 49.1: 69-77.
- [128] ZHAO, Xu, et al. A New Parameter Estimator for the Generalized Pareto Distribution under the Peaks over Threshold Framework. *Mathematics*, 2019, 7.5: 406.

- [129] PARK, Myung Hyun; KIM, Joseph HT. Estimating extreme tail risk measures with generalized Pareto distribution. *Computational Statistics & Data Analysis*, 2016, 98: 91-104.
- [130] DE ZEA BERMUDEZ, P.; KOTZ, Samuel. Parameter estimation of the generalized Pareto distribution—Part I. *Journal of Statistical Planning and Inference*, 2010, 140.6: 1353-1373.
- [131] HALD, Anders. *A History of Mathematical Statistics from 1750 to 1930*. New York: Wiley, 1998.
- [132] MONTGOMERY, Douglas C.; RUNGER, George C. *Applied Statistics and Probability for Engineers*. Wiley, 2007.
- [133] SMITH, Richard L. Maximum likelihood estimation in a class of nonregular cases. *Biometrika*, 1985, 72.1: 67-90.
- [134] GRIMSHAW, Scott D. Computing maximum likelihood estimates for the generalized Pareto distribution. *Technometrics*, 1993, 35.2: 185-191.
- [135] MACKEY, Michael C.; GLASS, Leon. Oscillation and chaos in physiological control systems. *Science*, 1977, 197.4300: 287-289.
- [136] SPANGENBERG, Mariana, et al. Detection of variance changes and mean value jumps in measurement noise for multipath mitigation in urban navigation. *Navigation*, 2010, 57.1: 35-52.
- [137] L'ECUYER, Pierre. History of uniform random number generation. In: *2017 Winter Simulation Conference (WSC)*. IEEE, 2017. p. 202-230.
- [138] MAURYA, Mano Ram; RENGASWAMY, Raghunathan; VENKATASUBRAMANIAN, Venkat. Fault diagnosis using dynamic trend analysis: A review and recent developments. *Engineering Applications of artificial intelligence*, 2007, 20.2: 133-146.
- [139] EGAN, James P. *Signal detection theory and ROC-analysis*. Academic press, 1975.
- [140] FAWCETT, Tom. An introduction to ROC analysis. *Pattern recognition letters*, 2006, 27.8: 861-874.
- [141] GERTLER, Janos. *Fault detection and diagnosis in engineering systems*. CRC press, 1998.
- [142] YU, Kangqing, et al. Real-time Outlier Detection over Streaming Data. In: *2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People*

- and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCCom/IOP/SCI)*.
IEEE, 2019. p. 125-132.
- [143] RAMEZANI, Ramin; ANGELOV, Plamen; ZHOU, Xiaowei. A fast approach to novelty detection in video streams using recursive density estimation. In: *2008 4th International IEEE Conference Intelligent Systems. IEEE, 2008*. p. 14-2-14-7.
 - [144] MARSLAND, Stephen; NEHMZOW, Ulrich; SHAPIRO, Jonathan. On-line novelty detection for autonomous mobile robots. *Robotics and Autonomous Systems*, 2005, 51.2-3: 191-206.
 - [145] NEHMZOW, Ulrich, et al. Novelty detection as an intrinsic motivation for cumulative learning robots. In: *Intrinsically Motivated Learning in Natural and Artificial Systems*. Springer, Berlin, Heidelberg, 2013. p. 185-207.
 - [146] VAN ROSUM, G.; DRAKE, F. L. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.
 - [147] OLIPHANT, Travis E. *A guide to NumPy (Vol. 1)*. Trelgol Publishing USA, 2006.
 - [148] VIRTANEN, P. et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 2020, 17(3), 261-272.
 - [149] MAREŠ, Jan, et al. Process state classification of fed-batch fermentation based on process variables analysis. *Biochemical Engineering Journal*, 2016, 112: 178-185.

Příloha

A Vybrané publikace

Introduction to Extreme Seeking Entropy

Jan Vrba ^{1,*} and Jan Mareš ^{1,2,*}

¹ Department of Computing and Control Engineering, Faculty of Chemical Engineering, University of Chemistry and Technology, 166 28 Prague, Czech Republic

² Department of Process Control, Faculty of Electrical Engineering and Informatics, University of Pardubice, 530 02 Pardubice, Czech Republic

* Correspondence: Jan.Vrba@vscht.cz (J.V.); Jan.Mares@vscht.cz (J.M.)

Received: 9 December 2019; Accepted: 8 January 2020; Published: 12 January 2020



Abstract: Recently, the concept of evaluating an unusually large learning effort of an adaptive system to detect novelties in the observed data was introduced. The present paper introduces a new measure of the learning effort of an adaptive system. The proposed method also uses adaptable parameters. Instead of a multi-scale enhanced approach, the generalized Pareto distribution is employed to estimate the probability of unusual updates, as well as for detecting novelties. This measure was successfully tested in various scenarios with (i) synthetic data, (ii) real time series datasets, and multiple adaptive filters and learning algorithms. The results of these experiments are presented.

Keywords: novelty detection; learning system; learning; time series; learning entropy; extreme seeking entropy

1. Introduction

Novelty detection (ND) plays an important role in signal processing. Many research groups have dealt with both the methods and applications because there are many complex tasks where accurate ND is needed. However, the success of this method depends on the type of data, so the current methods usually give good performance and results only for specific datasets. As more data are being analyzed currently, there is a greater need for new methods of ND. Furthermore, the increasing computational power provides more possibilities and methods that were not possible to use a few decades ago, but can now be performed for real-time tasks easily. For these reasons, we consider the topic of ND to be vital.

Two different approaches have been established over the last few decades. The first approach is based on the statistical features of the data [1], and some methods also use extreme value theory to estimate the novelty of the data [2–5]. The second approach uses learning systems [6–8]: the attributes of a learning system are used to obtain information about novelties in the data. Over the last decade, many new methods have been proposed in the field of machine learning [9]. The set membership algorithm [10–12] uses the prediction error for better accuracy, reducing the computational resources required and assuring a greater robustness with the proper filter, especially for data without drift. Bukovsky et al. proved that the learning effort of a learning system can be used to estimate a measure of the novelty for each data point [13,14], but a shortcoming of that method is that it is hard to interpret the ND score. A similar approach, combining the prediction error with adaptive weight increments, was proposed in [15]. That method also lacks the possibility of a meaningful interpretation of the ND score. It was also already shown that the accuracy of the learning system is not necessarily correlated with the accuracy of the ND [16] and that simple predictors are useful even for signals that are produced by complex systems (e.g., EEG, ECG).

ND brings a new point of view to complex signal analysis. Research groups have started dealing with the early diagnosis of different diseases where ND plays an important role. Taoum et al. presented

ND and data fusion methods to identify acute respiratory problems [17]. Rad introduced ND for gait and movement monitoring to diagnosis Parkinson's disease and autism spectrum disorders [18]. Burlina used ND algorithms in the diagnosis of different muscle diseases [19].

Other fields where ND can be found are information and mechanical engineering. Hu introduced ND as an appropriate tool for monitoring the health of mechanical systems, where it is usually impossible to know every potential fault [20]. Surace described the application of ND to the simulation of an offshore steel platform [21].

In this article, a new method for ND is introduced. The proposed method combines both a statistics based approach and a learning systems based approach. The changes of the adaptive parameters of the learning system obtained via an incremental learning algorithm are evaluated. A new measure, called extreme seeking entropy, is then estimated. It is shown that the proposed measure corresponds to different types of novelties in various datasets and how it may be useful for diagnostics and failure detection tasks. It also outperforms the other unsupervised adaptive ND methods.

This paper is organized as follows. Section 2 describes the specifications of the learning system and learning algorithm used during the experiments. Section 3 recalls the learning entropy algorithm and an error and learning based novelty detection method. Then, the general suitable properties of learning based information are discussed. Section 4 introduces the new measure of novelty, and the ND algorithm based on this measure is presented. Section 5 describes a case study where both synthetic and real datasets are used to show the usability of the proposed algorithm and also contains the rationale behind the selection of the experiments. Section 6 contains the rate detection of the proposed algorithm in two cases, namely detection of a change in the trend and the detection of a step change of a signal generator. The last two sections are dedicated to limitations and further challenges, Section 7, and then our conclusions, Section 8.

2. Review of the Learning Systems Used

All the supervised learning systems used in the experimental analysis are introduced in this section. In general, assume that the output of the learning system is a function of weights and the input data:

$$y = f(\mathbf{w}, \mathbf{x}) \quad (1)$$

where $y \in R$ denotes the output, $\mathbf{w} \in R^n$ is the vector of its adaptable parameters, $\mathbf{x} \in R^n$ is a vector that contains the input data, and f is the mapping function that maps the input data and weights to the output. The following adaptation is done in order to minimize the error:

$$e(k) = d(k) - y(k), \quad (2)$$

where k is a discrete time index and $d(k) \in R$ is the target of the supervised learning system (the desired output). The update of the weights \mathbf{w} is done with every new sample as follows:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \Delta\mathbf{w}(k) \quad (3)$$

where $\Delta\mathbf{w} \in R^n$ is a vector that contains the updates of the adaptive parameters. This update depends on the learning algorithm used. The learning algorithms will be discussed later.

2.1. Adaptive Models

The adaptive models used during the experiments are described briefly in this section.

2.1.1. Linear Adaptive Filter

One of the simplest adaptive models is the linear adaptive filter, also known as the linear neural unit (LNU), with finite impulse response (FIR). The output of this model at a discrete time index k is given by:

$$y(k) = \sum_{i=1}^n w_i \cdot x_i(k), \quad (4)$$

which is equivalent to

$$y(k) = \mathbf{w}^T(k) \cdot \mathbf{x}(k), \quad (5)$$

where $\mathbf{w}^T(k) = [w_1(k), w_2(k), \dots, w_n(k)] \in R^n$ is the row vector of adaptive weights and $\mathbf{x}^T(k) = [x_1(k), x_2(k), \dots, x_n(k)] \in R^n$ is the column input vector. The vector of adaptive weights is updated with every new sample obtained, and the size of the update depends on the learning algorithm used. In general, \mathbf{x} may contain the history of a single input or even the history of multiple inputs.

2.1.2. An Adaptive Filter Based on Higher Order Neural Units

The quadratic neural unit (QNU) [22–24] (also known as a second order neural unit) is a non-linear predictive model. The output of the QNU is:

$$y(k) = \sum_{i=0}^n \sum_{j=i}^n w_{i,j} \cdot x_i(k) \cdot x_j(k) \quad (6)$$

where often, $x_0 = 1$. This is equivalent to:

$$y(k) = \mathbf{w} \cdot \mathbf{colx}, \quad (7)$$

where the column input vector \mathbf{colx} for n inputs has the general form:

$$\mathbf{colx} = [1, x_1, \dots, x_n, x_1^2, x_1 \cdot x_2, \dots, x_1 \cdot x_n, x_2^2, x_2 \cdot x_3, \dots, x_2 \cdot x_n, \dots, x_{n-1} \cdot x_n, x_n^2]^T \quad (8)$$

and \mathbf{w} is a row vector of adaptive weights that has the same length as \mathbf{colx} . Note that the first term in \mathbf{colx} , $x_0 = 1$, should be used when the data have a non-zero offset.

2.2. Learning Algorithms

To prove the generality of the adaptive weight evaluation approach for novelty detection, different learning algorithms have been tested. Both algorithms are heavily used in signal processing and machine learning.

2.2.1. Normalized Least Mean Squares Algorithm

The normalized least mean squares (NLMS) algorithm [25] is a variant of the least mean squares algorithm. The problem with the selection of the learning rate is solved by normalizing by the power of the input. It is a stochastic gradient approach. The update of this adaptive algorithm is given by:

$$\Delta \mathbf{w}(k) = \frac{\mu \cdot \mathbf{x}(k) \cdot e(k)}{\epsilon + \mathbf{x}^T(k) \cdot \mathbf{x}(k)}, \quad (9)$$

where $\epsilon \in R$ is a small positive constant used to avoid division by zero, $\mu \in R$ is the learning rate, and $e \in R$ is the error defined as in (2). According to the normalization of the learning rate shown in (9), it is necessary to choose a learning rate μ satisfying $0 \leq \mu \leq 2$ to preserve the stability of the NLMS algorithm.

2.2.2. Generalized Normalized Gradient Descent

The generalized normalized gradient descent (GNGD) [26] algorithm is another algorithm for linear adaptive FIR filters. Due to its adaptation of the learning rate based on the signal dynamics, it converges in places where the NLMS algorithm diverges. The update of this adaptive algorithm is given by:

$$\Delta w(k) = \eta(k)e(k)\mathbf{x}(k) \quad (10)$$

with:

$$\eta(k) = \frac{\mu}{\|\mathbf{x}(k)\|_2^2 + \epsilon(k)}$$

$$\epsilon(k) = \epsilon(k-1) - \rho\mu \frac{e(k)e(k-1)\mathbf{x}^T(k)\mathbf{x}(k-1)}{(\|\mathbf{x}(k-1)\|_2^2 + \epsilon(k-1))^2}$$

where $\eta \in R$ is the adaptive learning rate, $\epsilon \in R$ is a compensation term, and ρ is the step size adaptation parameter, which should be chosen so as to satisfy $0 \leq \rho \leq 1$.

3. On the Evaluation of the Increments in the Adaptive Weights in Order to Estimate the Novelty in the Data

This section recalls two ND methods that evaluate the increments in the adaptive weights, namely learning entropy, and error and learning based novelty detection. Those methods are compared with the proposed algorithm in Sections 4 and 6. Then, the general properties of the learning based information measure will be discussed.

3.1. Learning Entropy: A Direct Algorithm

The recent publication on Learning Entropy [14] specifies a direct algorithm to estimate the learning entropy (LE) as follows.

$$LE(k) = \sum_{i=1}^{n_w} z(|\Delta w_i(k)|) \quad (11)$$

Here, z is a special Z-score, given as follows:

$$z(|\Delta w_i(k)|) = \frac{|\Delta w_i(k)| - \overline{|\Delta \mathbf{w}_i^M(k-1)|}}{\sigma(|\Delta \mathbf{w}_i^M(k-1)|)} \quad (12)$$

where $\overline{|\Delta \mathbf{w}_i^M(k-1)|}$ is the mean of the last M increments of w_i , $\sigma(|\Delta \mathbf{w}_i^M(k-1)|)$ is their standard deviation, and n_w is the number of adaptive weights. According to Equation (15), the function f in this case corresponds to the special Z-score function z , and the function A is represented by the sum over the adaptive weights.

3.2. Error and Learning Based Novelty Detection

Another recently published method that evaluates the increments of the adaptive weight together with the prediction error is ELBND [15]. ELBND describes every sample with the value obtained as follows:

$$ELBND(k) = \max_{1 \leq i \leq n_w} |\Delta w_i(k) \cdot e(k)| \quad (13)$$

or, alternatively,

$$ELBND(k) = \sum_{i=1}^{n_w} |\Delta w_i(k) \cdot e(k)|. \quad (14)$$

In this case, the function f is represented by multiplying the i th adaptive weight increment Δw_i by the prediction error e . The function A is the maximum of the vector in the case of ELBND given by Equation (13) and the sum over the weights in the case of the ELBND given by Equation (14).

3.3. General Properties of A Suitable Learning Based Information Measure

Learning entropy was proposed in [13,14]. It is a learning based information measure L that, in general, evaluates unusually large learning increments, as follows:

$$L(k) = A(f(\Delta \mathbf{w}(k))) \quad (15)$$

where A is a general aggregation function and f is a function that quantifies the irregularity in the learning effort [14].

Another form for f and A will be presented in the present paper. Firstly, the function f is presented.

Assume that the value of f should be high when the increments $\Delta \mathbf{w}$ are unusually high. Furthermore, this function also takes the history of those increments as input. As stated, some cumulative distribution function of each weight increment seems suitable. This cumulative distribution function (cdf) is discussed later in this paper. The question is how to deal with the aggregation function A . Under the assumption that each weight is independent of the others, it is possible to choose the aggregation function A as follows:

$$A(f(\Delta \mathbf{w}(k))) = -\log \prod_{i=1}^n (1 - f_{cdf_i}(|\Delta w_i(k)|)). \quad (16)$$

The function A in the stated form is high for high cdf values of the weight updates, and hence for the values where the cdf is close to one. The function $1 - f_{cdf_i}$ can be viewed as the complementary cumulative distribution function (or the survival function, also known as the reliability function). This approach clearly avoids the need for a multi-scale approach. The result is that much fewer parameters are needed for detecting potential novelties. Only the crucial choice of the cdf remains. In the next section, a suitable probability distribution will be presented, together with the new novelty detection algorithm.

4. Extreme Seeking Entropy

4.1. The Generalized Pareto Distribution

A normal distribution is used in some novelty detection algorithms [27–29]. However, the normal distribution cannot always be used, especially when the description of the data by a mean and a symmetric range of variation would be misleading [30]. Let us mention the Pickands–Balkema–de Haan theorem [31,32], which states that if we have a sequence X_1, X_2, \dots of independent and identically distributed random variables and F_u is their conditional excess distribution function (over the threshold u), then:

$$F_u(x) \rightarrow GPD(\xi, \mu, \sigma)(x), \text{ as } u \rightarrow \infty \quad (17)$$

where GPD is the generalized Pareto distribution and F_u is defined by:

$$F_u(x) = P(X - u \leq x, X > u) = \frac{F(u+x) - F(u)}{1 - F(u)} \quad (18)$$

for $0 \leq x \leq x_F - u$, where x_F is the right endpoint of the underlying unknown distribution F . The probability density function of the GPD takes the form:

$$f_{(\xi, \mu, \sigma)}(x) = \begin{cases} \frac{1}{\sigma} \left(1 + \frac{\xi(x-\mu)}{\sigma} \right)^{\left(-\frac{1}{\xi}-1\right)} & \text{for } \xi \neq 0, \\ \exp\left(-\frac{x-\mu}{\sigma}\right) & \text{for } \xi = 0. \end{cases} \quad (19)$$

where in general, $\mu \in (-\infty, +\infty)$ is a location parameter, $\sigma \in (0, \infty)$ is the scale, and $\xi \in (-\infty, \infty)$ is a shape parameter. The corresponding cumulative distribution function then takes the form:

$$F_{(\xi, \mu, \sigma)}(x) = \begin{cases} 1 - \left(1 + \frac{\xi(x-\mu)}{\sigma} \right)^{-\frac{1}{\xi}} & \text{for } \xi \neq 0, \\ 1 - \exp\left(-\frac{x-\mu}{\sigma}\right) & \text{for } \xi = 0. \end{cases} \quad (20)$$

Note that the support is $x \geq \mu$ if $\xi \geq 0$, and $\mu \leq x \leq \mu - \sigma/\xi$ if $\xi < 0$ where $\mu \in R$, $\sigma > 0$, and $\xi \in R$. In Figure 1, we show the ability of the GPD to deal with many possible shapes of the tails of the distributions. Note that if $\xi = 1$, it is equivalent to the uniform distribution; if $\xi = 0$, it is equivalent to the exponential distribution; if $\xi = -0.5$, it is the triangular distribution; if $-0.5 < \xi < 0$, it is a light tailed distribution (e.g., the normal distribution or the Gumbel distribution); if $\xi > 0$, it is a heavy tailed distribution (e.g., the Pareto distribution, the log-normal distribution, or Student's t -distribution); and if $\xi < -1$, it is a monotonically increasing distribution with compact support (e.g., the beta distribution).

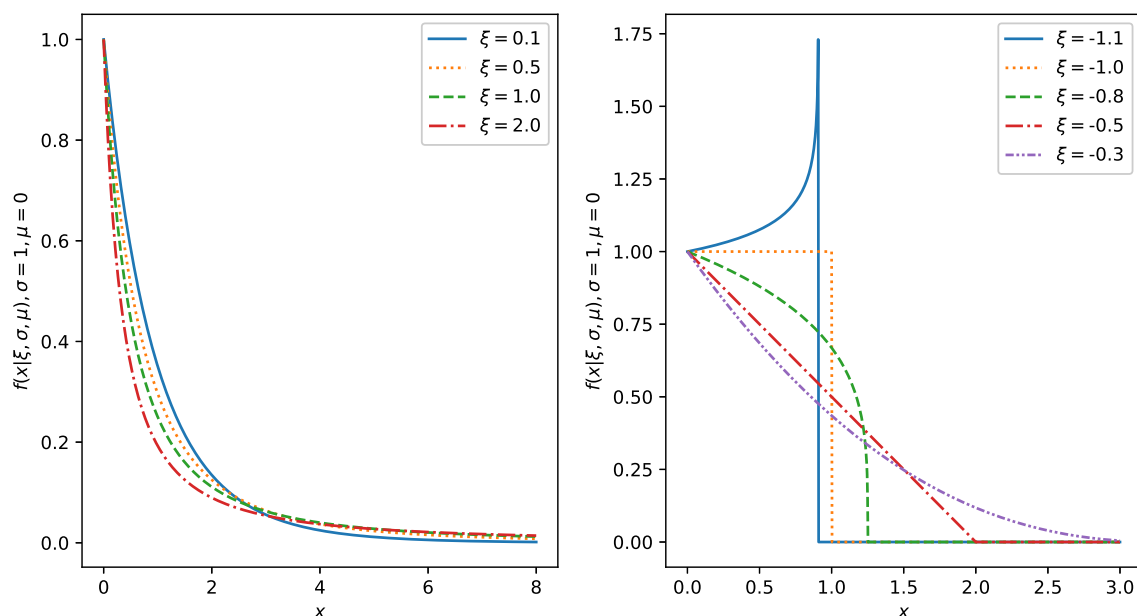


Figure 1. GPD probability density function with various parameters ξ , and fixed parameters $\sigma = 1$, $\mu = 0$.

As long as we do not know the distribution of increments of the adaptive weights, it is appropriate to use the GPD due to its universality in modeling the tails of other distributions [33–35]. As the aim is to evaluate unusually high increments of an adaptive system, the need for some threshold arises: denote this threshold by z . This threshold should divide the weight increments into two sets. An increment that is lower than the threshold should belong to the set that contains the usual high

increments; denote this by L . However, an increment that is greater than or equal to this threshold should belong to the set H . Assume that both sets exist for every adaptable parameter, so for the i th adaptable parameter w_i , we should set a threshold z_i so the weight updates belong to the sets as follows.

$$\forall |\Delta w_i| < z_i \in L_i \quad (21)$$

$$\forall |\Delta w_i| \geq z_i \in H_i \quad (22)$$

The increments belonging to L_i will be unlikely to contain any information about a novelty in the adaptation, so we are not going to evaluate them. The set H_i should contain the weight increments that are drawn from the GPD if the choice of the threshold was appropriate. The threshold z_i depends on the method chosen, peaks over threshold, which will be discussed in the following subsection.

4.2. The Peaks over Threshold Method

The main issue in GPD fitting is the estimation of a suitable threshold, z . If the threshold is too high (i.e., there are only a few points that exceed it), then the parameters of the GPD suffer a high variance. If the threshold is too low, then the GPD approximation is not reliable. Therefore, the proper choice of threshold is crucial for the performance of the ND algorithm. There are many approaches to estimating the threshold [36]. To show the usability of the proposed ND algorithm, multiple rules of thumb [37–39] for the choice of the threshold have been used. Let l be the number of samples used for the GPD fitting and n_s be the total number of samples available:

$$l_1 = \lceil 0.1 \cdot n_s \rceil \quad (23)$$

$$l_2 = \lceil \sqrt{n_s} \rceil \quad (24)$$

$$l_3 = \left\lceil \frac{\sqrt[3]{n_s^2}}{\log(\log(n_s))} \right\rceil \quad (25)$$

Note that we use the highest adaptive weight increment to estimate the GPD parameters. The peaks over threshold (POT) method is crucial for deciding whether $|\Delta w_k(k)|$ belongs to H_i or to L_i . In Section 5 are presented the results with different techniques of choosing the threshold.

4.3. Extreme Seeking Entropy Algorithm

In this subsection, the new novelty measure and the new novelty detection algorithm are presented. We will introduce the extreme seeking entropy measure, which is given as follows:

$$ESE(|\Delta \mathbf{w}(k)|) = -\log \prod_{i=1}^n (1 - f_{cdf_i}(|\Delta w_i(k)|)) \quad (26)$$

where:

$$f_{cdf_i}(|\Delta w_i(k)|) = \begin{cases} 0, & |\Delta w_i(k)| \in L_i \\ F_{(\xi_i, \mu_i, \sigma_i)}(|\Delta w_i(k)|), & |\Delta w_i(k)| \in H_i. \end{cases} \quad (27)$$

The proposed algorithm evaluates the value of ESE for every newly obtained weight increment. Note that if the weight increment is smaller than the threshold from the POT method, the addition to the novelty measure ESE is zero. Small probability increments, which are highly likely to contain a novelty, have a high value of ESE . To estimate the parameters of the GPD pdf, it is possible to process all available history samples, or only the n_s newest samples, with the POT method. The proposed algorithm is described by the following pseudocode (Algorithm 1).

Algorithm 1 Extreme seeking entropy algorithm.

```

1: set  $n_s$ , and choose the POT method
2: initial estimation of the parameters of the GPD:  $\xi_i, \mu_i, \sigma_i$  for each adaptable parameter
3: for each new  $d(k)$  do
4:   update the adaptive model to get  $\Delta \mathbf{w}(k)$ 
5:   proceed with the POT method
6:   if  $|\Delta w_i|(k) \in H_i$  then
7:     update the parameters  $\xi_i, \mu_i, \sigma_i$ 
8:   end if
9:   compute  $ESE$  according to (26)
10: end for

```

The proposed ND algorithm needs only one parameter to be set, which avoids the need for a multi-scale approach and overcomes the issues arising from setting multiple parameters. The parameter n_s can also take all available samples, if needed. Furthermore, there is the need to choose the proper POT method. Choosing the POT method depends strongly on the nature of the data. The limitation of the proposed method is the need for an initial estimate of the parameters of the GPD. We need a priori information about ξ , σ , and μ for each adaptive weight. If there are n_w adaptive weights, then we need $3 \cdot n_w$ parameters to start the extreme seeking entropy algorithm. If there is no a priori information about the parameters, we need at least n_s samples to obtain the first results. Another problem may arise if the type of underlying unknown distribution F or its parameters are significantly varying in time.

5. Experimental Results

5.1. The Design of the Experiments

The proposed ESE algorithm was studied in various testing schemes with synthetic data and with one real dataset. For each experiment, we also show the results of the ELBND and LE methods, for the sake of comparison. The parameter M that specifies the number of increments for the LE evaluation was set as $M = n_s$ in all experiments. The first experiment was the detection of perturbed data in the Mackey–Glass time series. This experiment was chosen due to the possibility of comparing it with the results published in [13]. The second experiment, with synthetic data, showed the ability of the ESE algorithm to detect a change in the standard deviation of the noise in a random data stream, which can be viewed as a novelty in the data. It was inspired by a problem that arises in hybrid navigation systems that use both GPS and dead-reckoning sensors [40]. The third experiment, involving a step change in the parameters of a signal generator, was an analogue to a problem that may arise in evaluating multiple stream random number generators [41], where we may detect and evaluate the probability of changes in the parameters of those generators. The fourth experiment was the detection of the disappearance of noise. This experiment was chosen as neither of the compared methods (LE, ELBND) were able to deal with this problem, where the disappearance of the noise could be also viewed as a novelty in the signal. The fifth experiment was the detection of a change in trend; this is a common problem in fault detection and diagnosis [42]. The last experiment was performed on the mouse EEG dataset. The aim of this experiment was to show that the proposed ESE algorithm was suitable even for real-world complex phenomena that are characterized by non-linear dynamics [43,44]. This dataset contained the start of an epileptic seizure, and we wanted to show that it was possible to detect this seizure with the proposed ESE algorithm. All of the experiments were carried out in the programming language Python [45], with the libraries Numpy [46], Scipy [47], and Padasip [48]. The graphs were plotted with the Matplotlib library [49]. The codes with the experiments can be obtained via email from the authors.

5.2. Mackey–Glass Time Series Perturbation

The first experiment was the detection of a perturbed sample in a deterministic chaotic time series. The time series data were obtained as the solution of the Mackey–Glass equation [50].

$$\frac{dy(t)}{dt} = \beta \cdot \frac{y(t-\tau)}{1 + y^\alpha(t-\tau)} - \gamma y(t) \quad (28)$$

with parameters $\alpha = 10$, $\beta = 0.2$, $\gamma = 0.1$, and $\tau = 17$. In all, 701 data samples were generated. The data sample at discrete time index $k = 523$ contained the perturbation, as follows:

$$y(523) = y(523) + 0.05 \cdot y(523) \quad (29)$$

The data series and detailed perturbation are depicted in Figure 2.

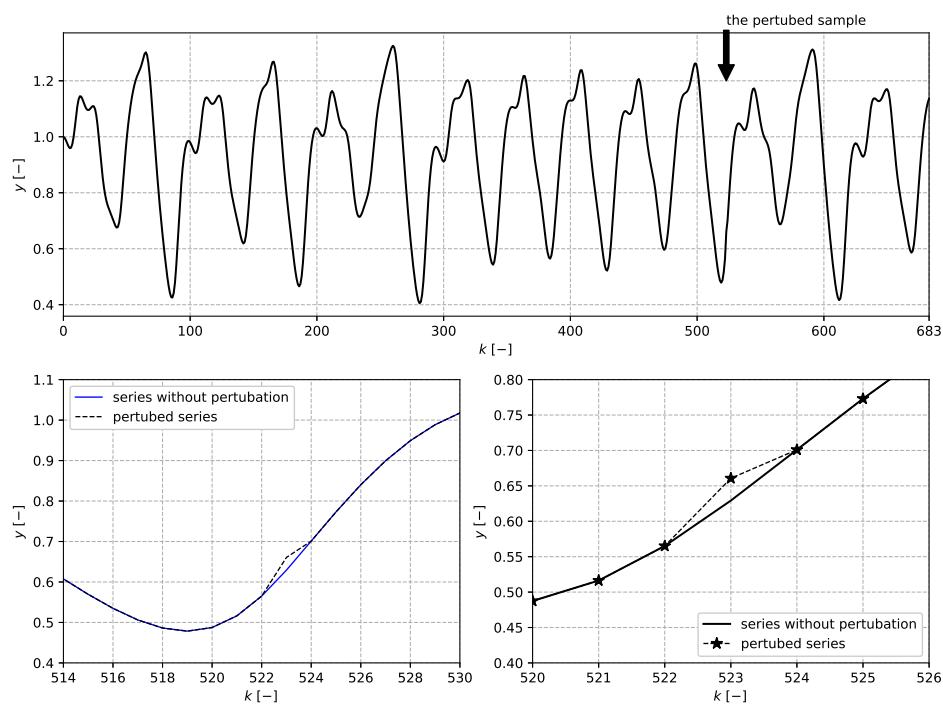


Figure 2. The (upper) plot displays an overview of the data with the perturbation. The (bottom) plots show the detailed perturbation.

The QNU was chosen for the data processing. The number of inputs to the QNU was set to $n = 4$, so the inputs are:

$$\mathbf{x} = [y(k-1), y(k-2), y(k-3), y(k-4)] \quad (30)$$

and hence, the adaptive filter had in all 15 adaptive weights. The parameters were updated with every newly obtained sample by means of the NLMS algorithm. The setting was the same as in [13]. The learning rate during the experiment was constantly set to $\mu = 1$. The POT method was chosen according to (23) with $n_s = 300$. The details of the adaptive filters and prediction error are depicted in Figure 3. The results of the ND are shown in Figure 4. Note that the global maximum in the ESE corresponds to the perturbed sample. The global maxima of the ELBND and LE methods correspond to the biggest prediction error, and not to the perturbed sample.

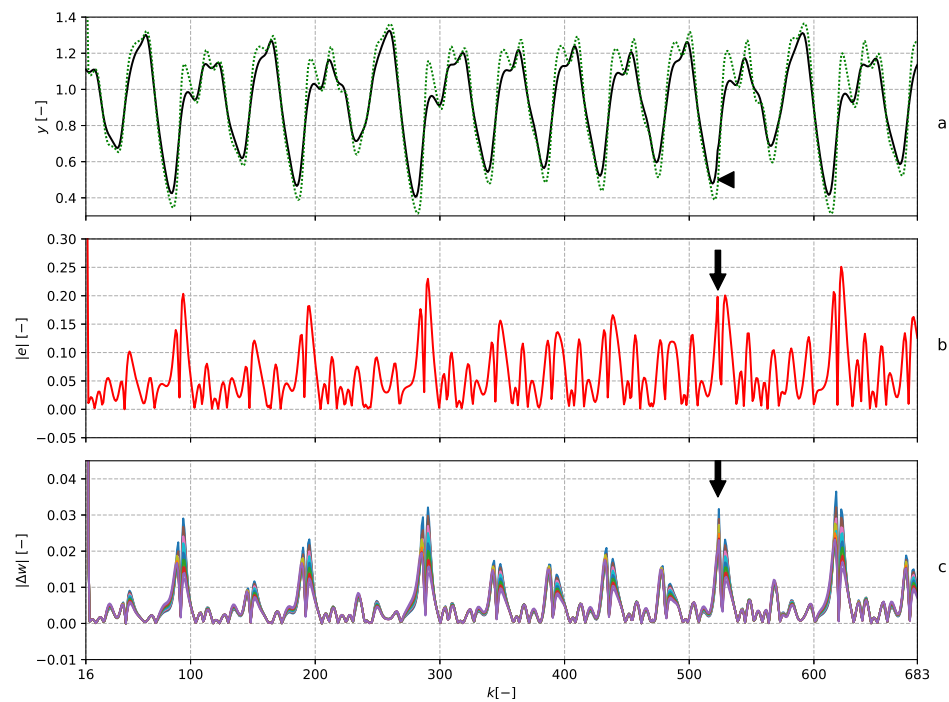


Figure 3. The graph (a) shows the data series with perturbation (solid black) and the output of the predictor (dotted green). The sample with perturbation is marked with the black arrow. The graph (b) shows the absolute value of the error of the predictor. The graph (c) shows the absolute value of the increments of the adaptable parameters.

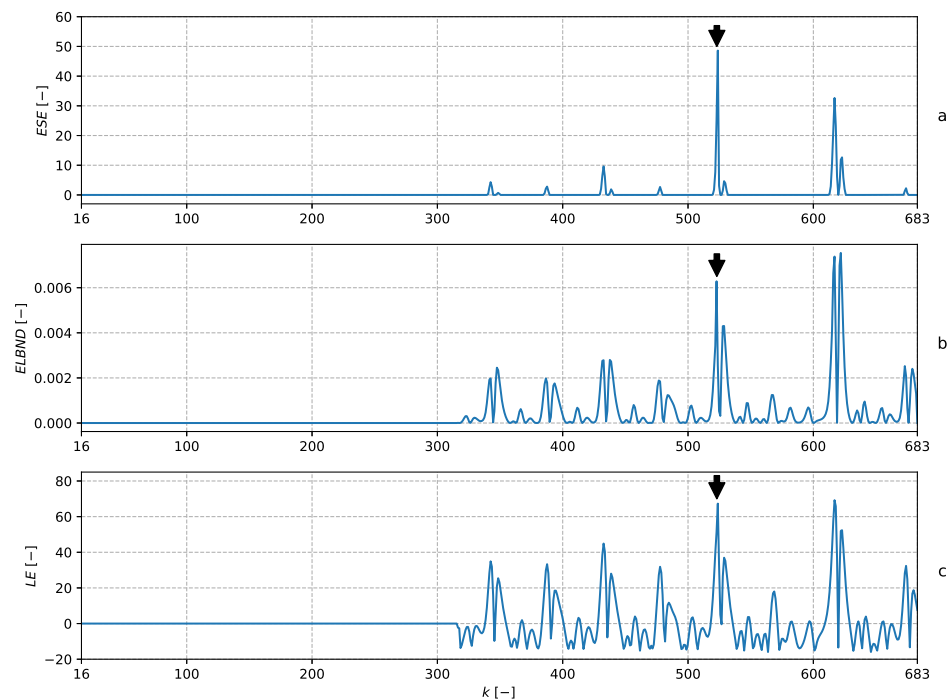


Figure 4. The graph (a) shows the ESE novelty score. Note that $n_s = 300$, so the first 300 samples are needed to obtain the first value of ESE. The graph (b) shows the ELBND novelty score. The first 300 samples are set to zero for easier comparison with the other methods. The graph (c) shows the LE novelty score.

5.3. Change of the Standard Deviation of the Noise in a Random Data Stream

The detection of a change in the standard deviation of the noise in the obtained data was carried out in the following experiment. Assume there are two inputs $x_1(k)$ and $x_2(k)$ and that the output $y(k)$ is related to them by:

$$y(k) = x_1(k) + x_2(k) + x_1(k) \cdot x_2(k) + v(k) \quad (31)$$

where $v(k)$ represents a Gaussian noise that is added to $w(k)$. The Gaussian noise has zero mean and standard deviation 0.1, $v \sim N(0, 0.1)$. The values of $x_1(k)$ and $x_2(k)$ are drawn from a uniform distribution, so that $x(k) \geq 0$ and $x(k) \leq 1$ for every k . At the discrete time index $k = 500$, the standard deviation of the noise changes to 0.2, so $v \sim N(0, 0.2)$. The QNU was chosen for the data processing. The number of inputs to the QNU was set to $n = 2$, so the inputs are:

$$\mathbf{x} = [x_1(k-1), x_2(k-1)] \quad (32)$$

and hence, the adaptive filter had three adaptive weights in all. The structure of the QNU corresponds to the structure of the data generator described by Equation (31). The parameters were updated with every newly obtained sample using the GNGD algorithm. The learning rate during the experiment was set to $\mu = 1$. The POT method was chosen according to (24) with $n_s = 500$. The results of the novelty detection and details about the adaptive filters are depicted in Figure 5. The a priori values of GPD for ESE and for LE were obtained using 500 samples, which are not shown in Figure 5. Note that the global maximum of the ESE corresponded to the change in standard deviation. The detection by the ELBND and LE was delayed.

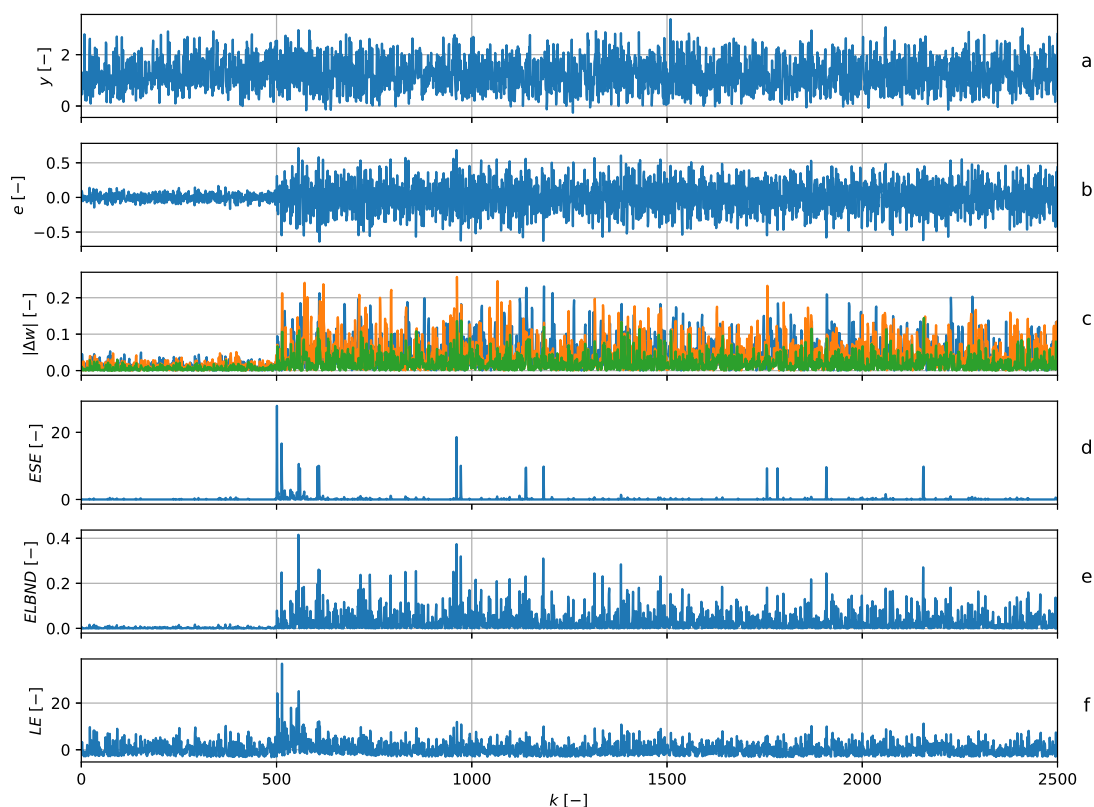


Figure 5. The graph (a) shows the data series (blue) and the output of the predictor (green). The graph (b) shows the error of the predictor. The graph (c) shows the absolute value of the increments of the adaptable parameters. The graph (d) shows the ESE novelty score. At discrete time index $k = 500$, there is a step change in the standard deviation of the noise and a corresponding global peak in ESE. Graphs (e) and (f) contain the results of the ELBND and LE methods.

5.4. Step Change in the Parameters of a Signal Generator

The scheme of this experiment was similar to the previous one. Assume there are two inputs $x_1(k)$ and $x_2(k)$ and one output $y(k)$, related by:

$$y(k) = x_1(k) + x_2(k) + x_1(k) \cdot x_2(k) + v(k) \quad (33)$$

where $v(k)$ represents a Gaussian noise that is added to $y(k)$. The Gaussian noise has zero mean and standard deviation 0.1, $v \sim N(0, 0.1)$. The values of $x_1(k)$ and $x_2(k)$ are drawn from a uniform distribution, so $x(k) \geq 0$ and $x(k) \leq 1$ for every k . At the discrete time index $k = 500$, the equation is changed to the following one:

$$y(k) = 0.4x_1(k) + 1.6x_2(k) + 0.99x_1(k) \cdot x_2(k) + v(k). \quad (34)$$

The QNU was chosen for the data processing. The number of inputs to the QNU was set to $n = 2$, so the inputs are:

$$\mathbf{x} = [x_1(k-1), x_2(k-1)] \quad (35)$$

and hence, the adaptive filter had three adaptive weights in all. Note that the structure of the QNU corresponded to the structure of the signal generator. The parameters were updated with every newly obtained sample, using the GNGD algorithm. The learning rate during the experiment was constantly set to $\mu = 1$. The POT method was chosen according to (23) with $n_s = 500$. The a priori values of GPD for ESE and for LE were obtained using 500 samples, which are not shown in Figure 6. The results of the novelty detection and details about the adaptive filters are depicted in Figure 6. Note that the ESE successfully detected the change in the parameters of the signal generator. The LE failed to detect this change, and the detection by ELBND was delayed. Furthermore, the value of the peak in ESE was significantly higher than that in the ELBND case.

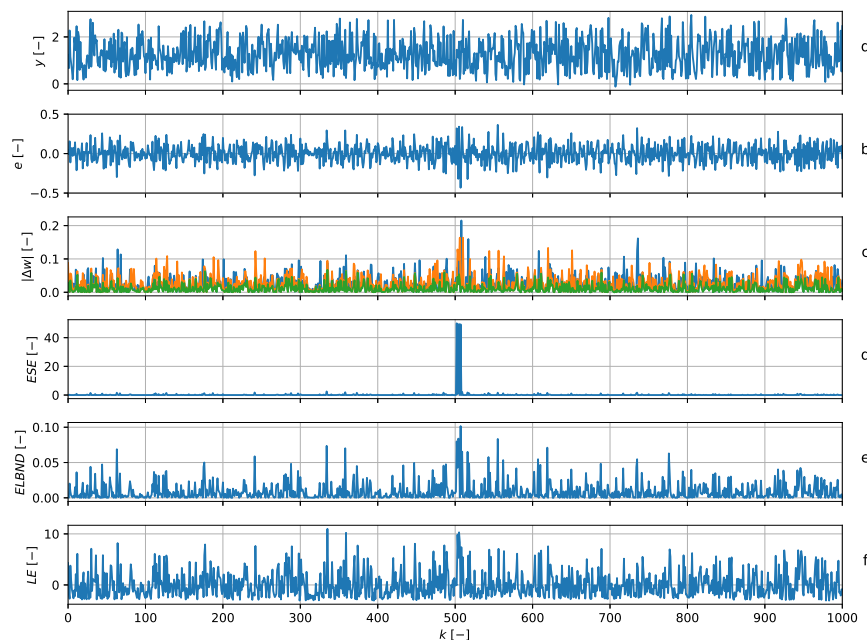


Figure 6. The graph (a) shows the data series (blue) and the output of the predictor (green). The graph (b) shows the value of predictors error. The graph (c) shows the absolute value of increments in the adaptable parameters. The graph (d) shows the ESE novelty score. At discrete time index $k = 500$, there is a step change in the parameters and a corresponding global maximum in ESE. Graphs (e) and (f) contain the results of the ELBND and LE methods. The detection with ELBND is delayed. The LE method failed to detect the change in this case, as the global maximum of LE is at $k = 338$.

5.5. Noise Disappearance

In this experiment, it was shown that the slightly reformulated algorithm could also deal with an immediate decrease of the learning effort. Assume that instead of an unusually high learning effort, we want to focus on an unusually low learning effort. The only change in the proposed algorithm was that we used the POT method to get l the smallest weight updates, and based on those, the parameters of the GPD would be estimated. The scheme of this experiment was similar to the previous one. We assumed there were two inputs $x_1(k)$ and $x_2(k)$ and one output $y(k)$, which were related by (31). However, in this case, at discrete time index $k = 500$, the noise was removed, so Equation (31) for $k \geq 500$ takes the form:

$$y(k) = x_1(k) + x_2(k) + x_1(k)x_2(k). \quad (36)$$

The QNU was chosen for the data processing. The number of inputs to the QNU was set to $n = 2$, so the inputs are:

$$\mathbf{x} = [x_1(k-1), x_2(k-1)] \quad (37)$$

and so, the adaptive filter had three adaptive weights in all. The structure of the adaptive filter was chosen to correspond to the structure of the signal generator. The parameters were updated with every newly obtained sample using the GNGD algorithm. The learning rate during the experiment was constantly set to $\mu = 1$. The POT method was chosen according to (23) with $n_s = 500$. Figure 7 shows that the peak in *ESE* corresponded to the disappearance of the noise. The *LE* and *ELBND* methods failed to detect the disappearance of the noise. For *ELBND*, these results were to be expected, as the values of the *ELBND* were high for a high prediction error and high adaptive weight increments.

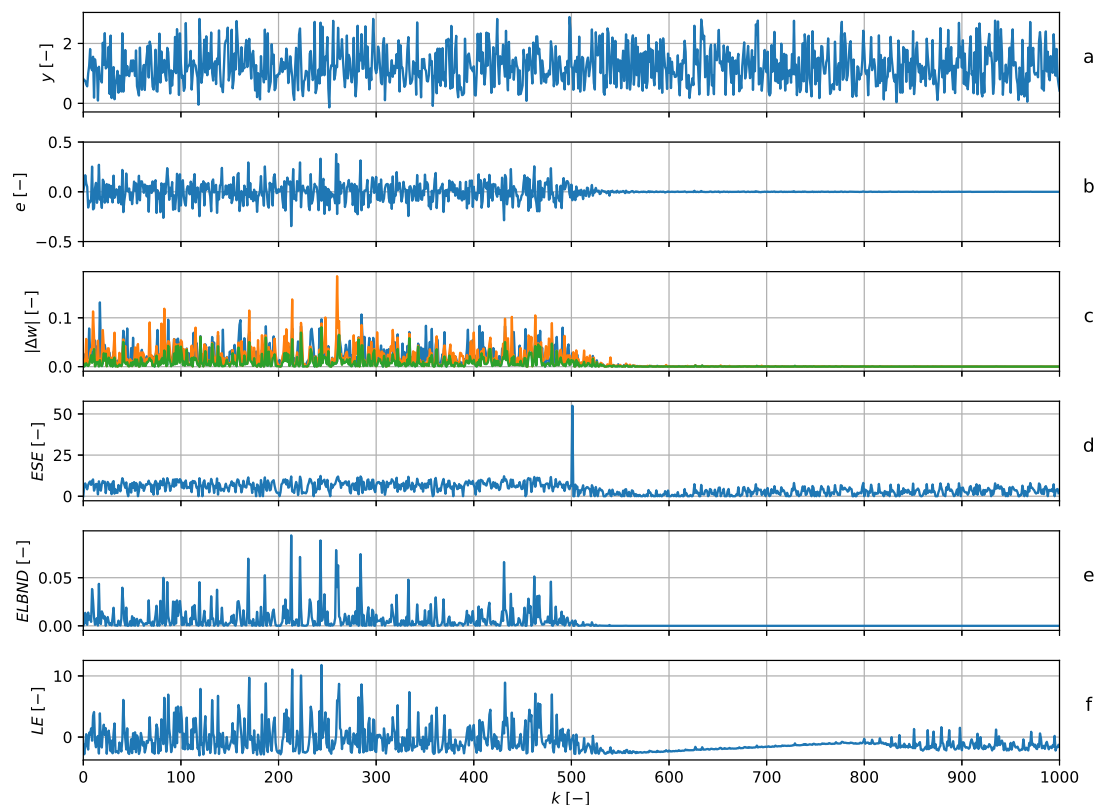


Figure 7. Noise disappearance detection. The graph (a) shows the data series (blue) and the output of the predictor (green). The graph (b) shows the error of the predictor. The graph (c) shows the absolute value of the increments of the adaptable parameters. The graph (d) shows the *ESE* novelty score. For the discrete time index $k \geq 500$, the noise is removed from the signal, which corresponds to the peak in *ESE*. Graphs (e) and (f) contain the results of the *ELBND* and *LE* methods.

5.6. Trend Change

The last experiment with artificial data was the detection of a change in trend. Assume that there are two inputs $x_1(k)$ and $x_2(k)$ and one output $y(k)$, related by:

$$y(k) = x_1(k) + x_2(k) + 0.01 \cdot k + v(k) \quad (38)$$

where $v(k)$ represents a Gaussian noise that is added to $y(k)$. The Gaussian noise had zero mean and standard deviation 0.1. At the discrete time index $k = 500$, there was a change in the trend, so Equation (38) changes to:

$$y(k) = x_1(k) + x_2(k) + 0.0105 \cdot k + v(k), \quad (39)$$

where $k \geq 500$. The LNU was chosen for the data processing. The number of inputs to the LNU was set to $n = 3$, so the inputs are:

$$\mathbf{x} = [x_1(k-1), x_2(k-1), 1] \quad (40)$$

and the adaptive filter had three adaptive weights in all. The structure of the adaptive filter was chosen in accordance with the structure of the signal generator. The parameters were updated with every newly obtained sample by means of the GNGD algorithm. The learning rate during the experiment was constantly set to $\mu = 1$. The POT method was chosen according to (23) with $n_s = 500$. Figure 8 shows that the peak in *ESE* corresponded to the trend change point, which was the same as the peak in *LE* and *ELBND*. Note that the value of the peak in *ESE* was significantly higher than in *LE* and *ELBND*.

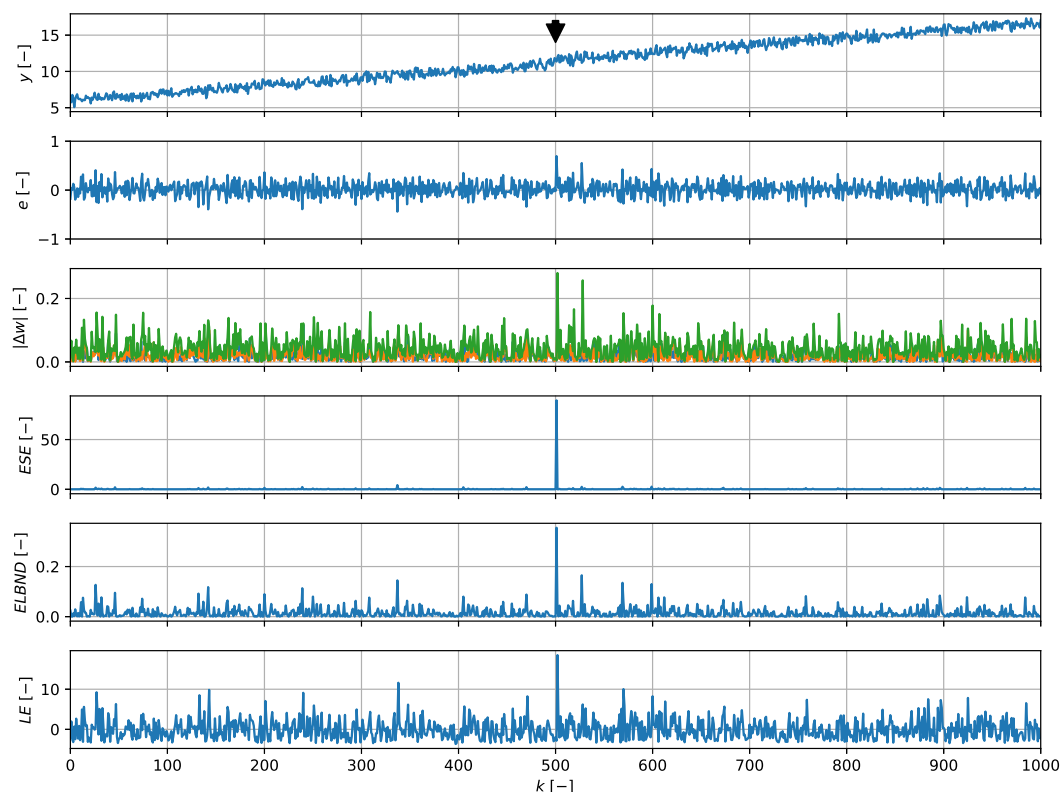


Figure 8. Trend change detection. The graph (a) shows the data series (blue) and the output of the predictor (green). The black arrow indicates the trend change. The graph (b) shows the error of the predictor. The graph (c) shows the absolute values of the increments in the adaptable parameters. The graph (d) shows the *ESE* novelty score. At discrete time index $k = 500$, there is a step change in the trend, which corresponds to the peak in *ESE*. Graphs (e) and (f) contain the results of the *ELBND* and *LE* methods and peaks corresponding to the trend change.

5.7. Detection of Epilepsy in Mouse EEG

The last experiment was with a mouse EEG signal. Three channels of the EEG data were chosen, which contained a significant seizure. According to the expert, the seizure started at about $k \approx 1700$, as is shown in Figure 9, which shows the z-scores of the EEG data.

The LNU was chosen for the data processing. The number of inputs to LNU was set to $n = 10$, so the inputs are:

$$\mathbf{x} = [x(k-1), x(k-2), \dots, x(k-10)] \quad (41)$$

and the adaptive filter had 10 adaptive weights in all. The number of inputs and filter structure were chosen experimentally. The parameters were updated with every newly obtained sample using the NLMS algorithm. The learning rate during the experiment was set to $\mu = 1$. The POT method was chosen according to (25) with $n_s = 1000$. Figure 10 shows that the peak in *ESE* approximately corresponded to the beginning of the seizure. Especially in channel C3, the peak in *ESE* was significant. The position of the peaks was at $k = 1735$ for channel C3, $k = 1698$ for channel Pz, and $k = 1727$ for channel Fp1.

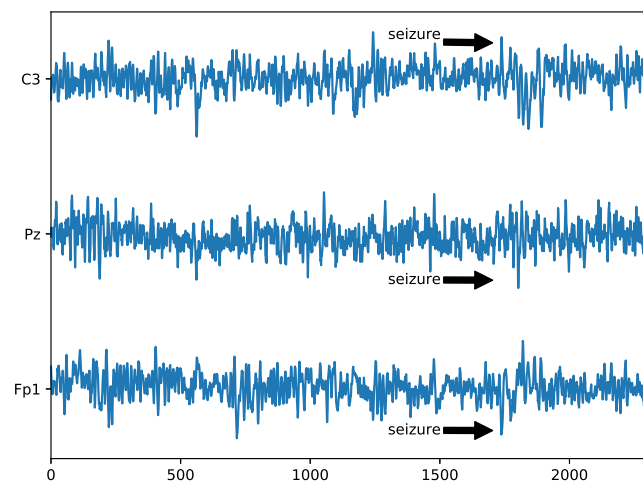


Figure 9. Selected mouse EEG channels with significant seizure. The data were standardized. The start of the seizure is approximately at discrete time index $k \approx 1700$.

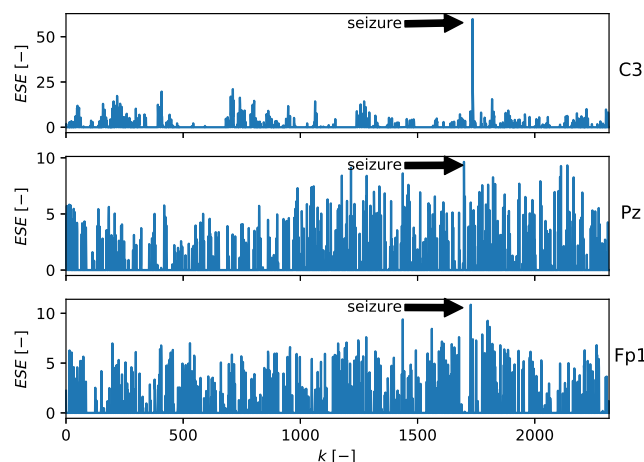


Figure 10. ESE value for mouse EEG data channels containing a seizure. The peaks approximately correspond to the beginning of the seizure. Note that channel C3 contains a significant peak in ESE compared to the other channels.

6. Evaluation of the ESE Detection Rate

This section is dedicated to evaluating the detection rate in two different cases. The first case was a step change in the parameters of a signal generator (similar to the experiment described in Section 5.4). The second case was the detection of a change in trend.

6.1. Step Change in the Parameters of a Signal Generator: Evaluation of the Detection Rate

Assume there are two inputs $x_1(k)$ and $x_2(k)$, one output $y(k)$, and weights a_1 , a_2 , and a_3 , related by:

$$y(k) = a_1 \cdot x_1(k) + a_2 \cdot x_2(k) + a_3 \cdot x_1(k) \cdot x_2(k) + v(k) \quad (42)$$

where $v(k)$ represents a Gaussian noise that is added to $y(k)$. The Gaussian noise had zero mean and standard deviation σ . The initial values of a_1 , a_2 , a_3 were drawn from the uniform distribution $U(-1, 1)$. At discrete time index $k = 200$, there was a step change in a_1 , a_2 , and a_3 , and their new values were drawn again from $U(-1, 1)$. The structure of the adaptive filter was the same as described in Section 5.4. The parameters were updated with every newly obtained sample using the GNGD algorithm. The POT method was chosen according to (23) with $n_s = 1200$. The performance of the ESE algorithm was compared with those of LE, ELBND, and plain prediction error evaluation. The a priori values of GPD for ESE and LE were obtained using 1200 samples with initial values for the parameters a_1 , a_2 , a_3 . For each experiment, the signal-to-noise ratio (SNR) was evaluated as follows:

$$SNR = 10 \log_{10} \frac{\sigma_s^2}{\sigma^2} \quad (43)$$

where σ_s is the standard deviation of the output of the system and σ is the standard deviation of the noise. The evaluation of the rate detection was performed as follows:

1. choose noise standard deviation σ
2. for given noise standard deviation σ , perform 1000 experiments, and at the beginning of each experiment, choose new parameters a_1 , a_2 , and a_3
3. successful detection was when the global peak in ESE, LE, ELBND, or prediction error was between discrete time index $k \geq 200$ and $k \leq 210$; compute the detection rate
4. compute the SNR for each experiment according to (43), and compute the average SNR for all experiments for given noise standard deviation σ

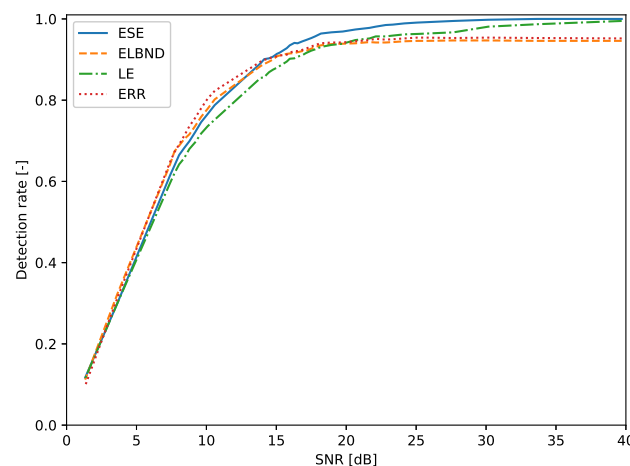


Figure 11. Detection of the step change in the parameters of a signal generator. The inputs of the signal generator are drawn from the uniform distribution $U(-1, 1)$. For $SNR > 15$ dB, the ESE algorithm outperforms in the detection rate the LE, ELBND, and error evaluation. For $SNR > 33$ dB, the ESE achieved a 100% detection rate.

The evaluation of the detection rate was performed for the inputs x_1, x_2 whose values were drawn from the uniform distribution $U(-1, 1)$ and from the normal distribution $N(0, 1)$. The results for the inputs drawn from the uniform distribution are depicted in Figure 11. The corresponding table with results for various SNRs is Table A2 (see Appendix A). The results for inputs drawn from the normal distribution are depicted in Figure 12. The corresponding table with results for various SNRs is Table A3 (see Appendix A).

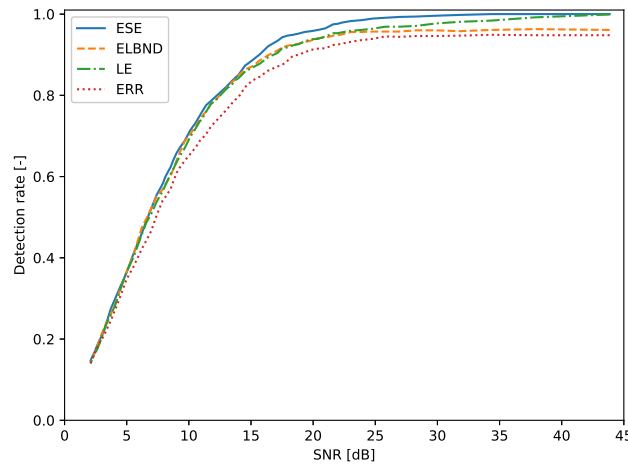


Figure 12. Detection of the step change in parameters of a signal generator. The inputs of the signal generator are drawn from the normal distribution $N(0, 1)$. For $SNR > 8$ dB, the ESE algorithm outperforms in the detection rate the LE, ELBND, and error evaluation. For $SNR > 34$ dB, the ESE achieved a 100% detection rate.

6.2. Detection of a Change in Trend: Evaluation of the Detection Rate

Assume there are two inputs $x_1(k)$ and $x_2(k)$ and one output $y(k)$, related by:

$$y(k) = x_1(k) + x_2(k) + 0.01 \cdot k + v(k) \quad (44)$$

where $v(k)$ represents a Gaussian noise that is added to $y(k)$. The Gaussian noise has zero mean and standard deviation σ . At discrete time index k , the trend changed, so the output of the system $y(k)$ for $k \geq 200$ is given by:

$$y(k) = x_1(k) + x_2(k) + (0.01 + a) \cdot k + v(k) \quad (45)$$

where a is drawn from the uniform distribution $U(-0.02, 0.02)$. The structure of the adaptive filter was the same as in the experiment described in Section 5.6. The parameters were updated with every newly obtained sample using the GNGD algorithm. The POT method was chosen according to (23) with $n_s = 1200$. The performance of the ESE algorithm was compared with LE, ELBND, and plain prediction error evaluation. The a priori values of the GPD for ESE and LE were obtained using 1200 samples where the output of the system was described by Equation (44). For each experiment, the SNR was evaluated according to (43). The evaluation of the rate detection was performed as follows:

1. choose noise standard deviation σ
2. for given noise standard deviation σ , perform 1000 experiments where at $k = 200$, there is a change in trend
3. successful detection is when the global peak in ESE, LE, ELBND, or prediction error is between discrete time index $k \geq 200$ and $k \leq 210$; compute the detection rate
4. compute the SNR for each experiment according to (43), and compute the average SNR for all experiments for given noise standard deviation σ

The evaluation of the detection rate was performed for inputs x_1, x_2 whose values were drawn from the uniform distribution $U(-1, 1)$. The results are depicted in Figure 13. The corresponding table with the results for various SNRs is Table A1 (see Appendix A).

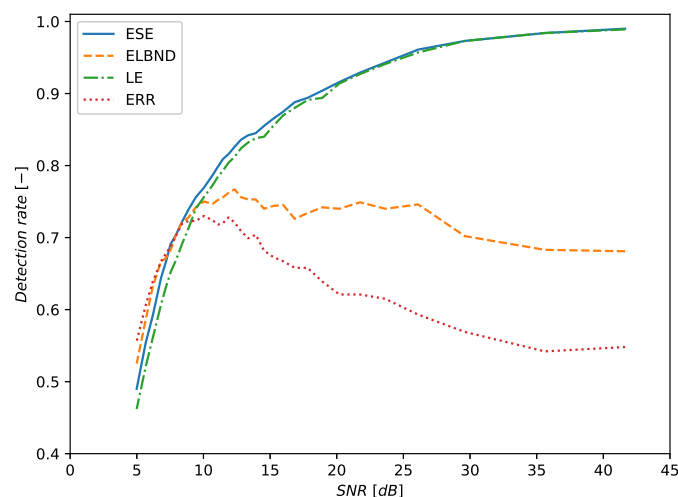


Figure 13. Detection of the trend change. The inputs of the signal generator are drawn from the uniform distribution $U(-1, 1)$. For $\text{SNR} > 8$ dB, the ESE algorithm outperforms in the detection rate the LE, ELBND, and error evaluation.

7. Limitations and Further Challenges

There is a significant limitation to using the ESE algorithm. As was already mentioned in Section 4, before we could obtain the first results, we needed to get a priori information about the parameters of the GPD or obtain a suitably large sample size to compute those parameters. This limitation arose from the nature of using the probability distribution and is common to many statistical approaches to ND. This was the main drawback compared to, e.g., the ELBND method, which was able to produce the results immediately. Another limitation of the presented algorithm is the selection of a suitable POT method, as the estimation of the parameters of the GPD and the selection of the threshold were strongly related to this. To avoid this issue, it was possible to implement some sophisticated parameter estimator that could deal with the optimal threshold selection (e.g., Zhang's method [51], an estimator based on generalized probability weighted moment equations [52], or a method that combines the method of moments and the likelihood moment [53]), but these are outside the scope of this article. Another challenge was how to combine the ESE of unusually low and unusually high increments together, because both could correspond to a novelty in the data. Further work will be oriented toward using adaptive filters whose adaptive parameters are non-linearly related to the output, e.g., fuzzy adaptive filters or non-linear adaptive Kalman filters. Furthermore, more learning algorithms should be tested. Another topic, which was not mentioned in this article, is that of deciding whether the value of the ESE implies a novelty in the data or not, so we need some threshold. To evaluate the precision of the classification, the area under the receiver operating characteristics [54,55] should be estimated. Due to the scope of this article, this was omitted, but it will be part of further work on the ESE.

8. Conclusions

This paper introduced a new measure of data novelty, called extreme seeking entropy, and a detection algorithm that used this measure. An experimental study was also presented. The algorithm evaluated the absolute value of the increments in the adaptive system weights that were unusually high. The generalized Pareto distribution was used to model those increments, and we tested whether a low probability of a weight increment corresponded to a novelty in the data. It was also shown

that the prediction error did not need to be correlated with a novelty in the data, so relatively simple, even inaccurate, adaptive models could be used. Five experiments with synthetic data including novelties and one experiment with a real mouse EEG signal were presented. It was shown that the proposed novelty detection algorithm was able to detect novelties in both kinds of data (real and synthetic) and that the proposed approach using simple adaptive models should be suitable for adaptive novelty detection. The detection rate of the proposed algorithm was evaluated for various SNRs in the scenarios of trend change detection and of a step change in the parameters of a signal generator. These scenarios were also tested with LE, ELBND, and prediction error evaluation. It was shown that for higher SNRs, the proposed ESE algorithm outperformed the other tested algorithms in terms of a successful detection rate in both scenarios.

Author Contributions: Conceptualization, J.V.; method, J.V.; software, J.V.; validation, J.V. and J.M.; investigation, J.V.; resources, J.V. and J.M.; writing, original draft preparation, J.V.; writing, review and editing, J.M.; visualization, J.V.; supervision, J.M.; project administration, J.M.; funding acquisition, J.M. All authors have read and agreed to the published version of the manuscript.

Funding: Financial support was from specific university research (MSMTNo 21-SVV/2019).

Acknowledgments: Jan Vrba would like to thank Matouš Cejnek for developing the PADASIP (Python Adaptive Signal Processing library) and Ivo Bukovský for helpful discussions about learning entropy and learning systems.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ND	novelty detection
ESE	extreme seeking entropy
LE	learning entropy
ELBND	error and learning based novelty detection
GEV	generalized extreme value
GPD	generalized Pareto distribution
GNGD	generalized normalized gradient descent
NLMS	normalized least mean squares
pdf	probability density function
cdf	cumulative density function
POT	peak over threshold
LNU	linear neural unit
QNU	quadratic neural unit
SNR	signal-to-noise ratio

Appendix A

Table A1. Trend change detection rates.

SNR (dB)	ESE (%)	ELBND (%)	LE (%)	Err (%)
41.63	99.0	68.1	98.9	54.8
35.62	98.4	68.3	98.4	54.2
29.61	97.3	70.2	97.3	56.9
26.11	96.1	74.6	95.7	59.3
23.64	94.3	74.0	94.1	61.5
21.74	92.9	74.9	92.7	62.1
20.20	91.6	74.0	91.4	62.1
18.92	90.4	74.2	89.4	63.9
17.81	89.4	73.4	89.1	65.8
16.85	88.8	72.6	88.0	65.8
16.00	87.5	74.5	87.0	66.7
15.23	86.5	74.4	85.5	67.3
14.55	85.5	74.0	84.0	68.2
13.92	84.5	75.3	83.8	70.4
13.35	84.2	75.3	83.2	69.9
12.82	83.6	75.6	82.4	70.9
12.34	82.6	76.7	81.2	72.1
11.88	81.6	76.2	80.4	72.8
11.46	80.9	75.6	79.3	72.0
11.07	79.8	75.2	78.4	71.8
10.70	78.7	74.7	77.3	72.4
10.02	76.9	75.0	75.6	73.0
9.42	75.6	74.2	74.0	72.3
8.88	73.9	72.8	71.3	72.3
8.39	72.2	72.1	69.1	71.9
7.95	70.4	70.4	66.9	70.3
7.54	69.1	68.3	65.2	68.8
7.16	66.5	67.1	63.0	67.7
6.82	64.4	66.7	60.7	66.7
6.20	59.2	63.2	56.1	63.9
5.67	55.4	58.6	52.1	60.6
4.99	49.0	52.5	46.2	55.6

Table A2. Step change detection rates for inputs drawn from uniform distribution $U(-1, 1)$.

SNR (dB)	ESE (%)	ELBND (%)	LE (%)	Err (%)
39.70	100.0	94.6	99.5	95.2
33.68	100.0	94.6	98.7	95.3
30.16	99.8	94.7	98.1	95.4
27.67	99.5	94.7	96.7	95.3
25.73	99.2	94.6	96.4	95.4
25.07	99.1	94.6	96.3	95.4
24.15	98.9	94.5	96.2	95.2
23.33	98.6	94.3	95.9	95.0
22.82	98.5	94.2	95.7	94.9
22.11	98.1	94.2	95.7	95.1
21.67	97.8	94.3	95.2	94.8
20.66	97.4	94.0	94.8	94.5
19.75	96.9	93.9	93.9	94.2
18.93	96.7	93.8	93.6	94.2
18.19	96.4	93.5	93.1	94.0
17.77	95.7	93.1	92.5	93.6
17.51	95.3	92.7	92.2	93.2
17.12	94.8	92.1	91.6	92.7
16.88	94.5	92.1	91.1	92.4
16.52	94.0	91.8	90.7	92.1
16.29	94.1	91.6	90.3	92.1
15.96	93.5	91.5	90.2	91.7
15.75	92.8	91.4	89.5	91.2
15.24	91.7	91.0	88.4	91.0
15.04	91.4	90.8	88.0	90.8
14.76	90.7	90.0	87.5	90.6
14.48	90.3	89.5	86.9	90.3
14.31	90.2	89.2	86.3	90.1
14.13	90.1	88.9	85.6	89.9
13.88	89.1	88.4	85.5	89.5
10.60	78.8	80.2	75.2	82.3
10.09	76.6	77.9	73.6	80.4
9.62	74.6	75.9	71.8	78.0
9.19	72.1	73.6	69.6	75.6
8.78	69.9	71.6	68.0	73.6
8.41	68.2	70.4	65.7	71.2
8.05	66.5	68.8	64.2	69.1
7.72	63.9	67.4	62.0	67.2
7.40	61.5	64.2	59.6	64.8
3.64	30.1	32.4	30.1	31.4
1.33	11.7	11.2	11.5	9.7

Table A3. Step change detection rates for inputs drawn from normal distribution $N(0, 1)$.

SNR (dB)	ESE (%)	ELBND (%)	LE (%)	Err (%)
43.86	100.0	96.1	99.9	94.8
37.84	100.0	96.3	99.2	94.8
34.32	100.0	96.1	98.4	94.9
31.83	99.8	95.8	98.1	94.7
29.89	99.6	96.0	97.7	94.6
28.31	99.4	96.0	97.1	94.6
26.97	99.3	95.7	96.9	94.4
25.82	99.1	95.7	96.9	94.5
24.80	98.9	95.7	96.4	93.9
23.89	98.5	95.6	96.1	93.6
23.06	98.3	95.6	95.9	93.2
22.31	97.9	95.0	95.5	92.7
21.96	97.6	94.8	95.3	92.5
21.62	97.5	94.7	95.3	92.4
20.98	96.5	94.2	94.5	91.6
20.39	96.1	94.1	93.9	91.5
19.84	95.8	93.5	93.8	91.2
19.32	95.6	93.1	93.3	90.6
18.83	95.2	92.7	92.6	90.0
18.37	94.9	92.5	92.3	89.7
17.93	94.7	92.2	91.8	88.5
17.52	94.3	91.8	91.6	87.5
17.12	93.4	91.0	90.4	87.1
16.39	92.1	89.9	89.4	86.1
15.71	90.1	88.6	87.8	84.6
15.09	88.6	87.1	86.8	83.6
14.52	87.3	86.4	85.8	82.0
13.98	85.0	84.9	84.2	79.5
13.73	84.3	83.9	83.7	78.9
13.48	83.6	83.3	82.7	78.1
13.02	82.1	81.7	81.6	76.4
12.16	79.7	78.9	78.8	73.6
11.77	78.6	77.8	77.9	72.3
11.40	77.6	76.1	76.0	70.5
11.05	75.9	74.8	74.1	69.3
10.88	75.0	74.1	73.6	68.8
10.55	73.3	72.4	71.9	67.2
10.09	71.3	70.4	69.7	65.5
9.67	68.8	68.7	67.0	63.9
9.27	67.1	65.9	65.0	62.2
9.01	65.8	64.0	63.5	60.9
8.77	64.2	62.0	62.0	59.0
8.54	62.4	60.2	60.7	57.0
8.10	59.9	58.1	57.8	54.9
7.89	58.2	56.8	56.4	54.1
7.41	55.5	55.1	53.8	50.5
6.97	52.1	52.2	50.8	46.6
6.57	49.0	49.6	48.6	44.2
6.20	45.9	47.3	45.6	42.1
5.54	41.0	40.7	40.1	37.8
4.98	36.3	36.5	36.5	34.7
4.50	33.0	32.4	32.3	30.6
4.08	30.0	28.8	28.4	27.3
3.71	27.4	26.0	25.9	24.5
3.39	24.2	24.1	23.9	22.3
3.10	21.6	22.2	21.4	20.5
2.85	19.9	20.3	19.2	19.4
2.63	18.5	18.1	17.5	17.8
2.43	16.9	16.5	16.3	16.6
2.25	15.8	15.4	15.3	15.2
2.09	14.7	14.3	14.0	14.0

References

1. Markou, M.; Singh, S. Novelty detection: A review—Part 1: Statistical approaches. *Signal Process.* **2003**, *83*, 2481–2497. [\[CrossRef\]](#)
2. Clifton, D.A.; Hugueny, S.; Tarassenko, L. Novelty detection with multivariate extreme value statistics. *J. Signal Process. Syst.* **2011**, *65*, 371–389. [\[CrossRef\]](#)
3. Hugueny, S.; Clifton, D.A.; Tarassenko, L. Probabilistic patient monitoring with multivariate, multimodal extreme value theory. In *International Joint Conference on Biomedical Engineering Systems and Technologies*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 199–211.
4. Clifton, D.A.; Clifton, L.; Hugueny, S.; Tarassenko, L. Extending the generalised Pareto distribution for novelty detection in high-dimensional spaces. *J. Signal Process. Syst.* **2014**, *74*, 323–339. [\[CrossRef\]](#)
5. Luca, S.; Clifton, D.A.; Vanrumste, B. One-class classification of point patterns of extremes. *J. Mach. Learn. Res.* **2016**, *17*, 6581–6601.
6. Markou, M.; Singh, S. Novelty detection: A review—Part 2: Neural network based approaches. *Signal Process.* **2003**, *83*, 2499–2521. [\[CrossRef\]](#)
7. Marsland, S. Novelty detection in learning systems. *Neural Comput. Surv.* **2003**, *3*, 157–195.
8. Polycarpou, M.M.; Trunov, A.B. Learning approach to nonlinear fault diagnosis: Detectability analysis. *IEEE Trans. Autom. Control* **2000**, *45*, 806–812. [\[CrossRef\]](#)
9. Pimentel, M.A.; Clifton, D.A.; Clifton, L.; Tarassenko, L. A review of novelty detection. *Signal Process.* **2014**, *99*, 215–249. [\[CrossRef\]](#)
10. Yazdanpanah, H.; Lima, M.V.; Diniz, P.S. On the robustness of set-membership adaptive filtering algorithms. *EURASIP J. Adv. Signal Process.* **2017**, *2017*, 72. [\[CrossRef\]](#)
11. Diniz, P.S.R. *Adaptive Filtering*; Springer: New York, NY, USA, 2013; doi:10.1007/978-1-4614-4106-9. [\[CrossRef\]](#)
12. Gollamudi, S.; Nagaraj, S.; Kapoor, S.; Huang, Y.F. Set-membership filtering and a set-membership normalized LMS algorithm with an adaptive step size. *IEEE Signal Process. Lett.* **1998**, *5*, 111–114. [\[CrossRef\]](#)
13. Bukovsky, I. Learning Entropy: Multiscale Measure for Incremental Learning. *Entropy* **2013**, *15*, 4159–4187. [\[CrossRef\]](#)
14. Bukovsky, I.; Kinsner, W.; Homma, N. Learning Entropy as a Learning-Based Information Concept. *Entropy* **2019**, *21*, 166. [\[CrossRef\]](#)
15. Cejnek, M.; Bukovsky, I. Concept drift robust adaptive novelty detection for data streams. *Neurocomputing* **2018**, *309*, 46–53. [\[CrossRef\]](#)
16. Bukovsky, I.; Kinsner, W.; Bila, J. Multiscale analysis approach for novelty detection in adaptation plot. In *Proceedings of the Sensor Signal Processing for Defence (SSPD 2012)*, London, UK, 25–27 September 2012; pp. 1–6.
17. Taoum, A.; Mourad-chehade, F.; Amoud, H. Early-warning of ARDS using novelty detection and data fusion. *Comput. Biol. Med.* **2018**, *102*, 191–199. [\[CrossRef\]](#) [\[PubMed\]](#)
18. Rad, N.M.; van Laarhoven, T.; Furlanello, C.; Marchiori, E. Novelty Detection using Deep Normative Modeling for IMU-Based Abnormal Movement Monitoring in Parkinson’s Disease and Autism Spectrum Disorders. *Sensors* **2018**, *18*, 3533. [\[CrossRef\]](#)
19. Burlina, P.; Joshi, N.; Billings, S.; Wang, I.J.; Albayda, J. Deep embeddings for novelty detection in myopathy. *Comput. Biol. Med.* **2019**, *105*, 46–53. [\[CrossRef\]](#)
20. Hu, L.; Hu, N.; Fan, B.; Gu, F. Application of novelty detection methods to health monitoring and typical fault diagnosis of a turbopump. *J. Phys. Conf. Ser.* **2012**, *364*, 012128. [\[CrossRef\]](#)
21. Surace, C.; Worden, K. A novelty detection method to diagnose damage in structures: An application to an offshore platform. In *The Eighth International Offshore and Polar Engineering Conference*; International Society of Offshore and Polar Engineers: Mountain View, CA, USA, 1998.
22. Bukovsky, I.; Homma, N.; Smetana, L.; Rodriguez, R.; Mironovova, M.; Vrana, S. Quadratic neural unit is a good compromise between linear models and neural networks for industrial applications. In *Proceedings of the 9th IEEE International Conference on Cognitive Informatics (ICCI’10)*, Beijing, China, 7–9 July 2010; pp. 556–560. [\[CrossRef\]](#)
23. Zhang, M. (Ed.) *Artificial Higher Order Neural Networks for Modeling and Simulation*; IGI Global: Hershey, PA, USA, 2013; doi:10.4018/978-1-4666-2175-6. [\[CrossRef\]](#)

24. Bukovsky, I.; Voracek, J.; Ichiji, K.; Noriyasu, H. Higher Order Neural Units for Efficient Adaptive Control of Weakly Nonlinear Systems. In *Proceedings of the 9th International Joint Conference on Computational Intelligence*; SciTePress—Science and Technology Publications: Setúbal, Portugal, 2017; doi:10.5220/0006557301490157. [CrossRef]
25. Haykin, S.S. *Adaptive Filter Theory*; Pearson Education India: Delhi, India, 2005.
26. Mandic, D.P. A generalized normalized gradient descent algorithm. *IEEE Signal Process. Lett.* **2004**, *11*, 115–118. [CrossRef]
27. Grubbs, F.E. Procedures for detecting outlying observations in samples. *Technometrics* **1969**, *11*, 1–21. [CrossRef]
28. Ma, J.; Perkins, S. Time-series novelty detection using one-class support vector machines. In *Proceedings of the International Joint Conference on Neural Networks*, 2003, Portland, OR, USA, 20–24 July 2003; Volume 3, pp. 1741–1745.
29. Ma, J.; Perkins, S. Online novelty detection on temporal sequences. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington, DC, USA, 24–27 August 2003; pp. 613–618.
30. Limpert, E.; Stahel, W.A. Problems with Using the Normal Distribution—And Ways to Improve Quality and Efficiency of Data Analysis. *PLoS ONE* **2011**, *6*, e21403. [CrossRef]
31. Pickands, J., III. Statistical inference using extreme order statistics. *Ann. Stat.* **1975**, *3*, 119–131.
32. Balkema, A.A.; De Haan, L. Residual life time at great age. *Ann. Probab.* **1974**, *2*, 792–804. [CrossRef]
33. Forbes, C.; Evans, M.; Hastings, N.; Peacock, B. *Statistical Distributions*; Wiley: Hoboken, NJ, USA, 2010.
34. Coles, S. *An Introduction to Statistical Modeling of Extreme Values*; Springer Series in Statistics; Springer: London, UK, 2013.
35. Lee, H.J.; Roberts, S.J. On-line novelty detection using the Kalman filter and extreme value theory. In *Proceedings of the 2008 19th International Conference on Pattern Recognition*, Tampa, FL, USA, 8–11 December 2008; doi:10.1109/icpr.2008.4761918. [CrossRef]
36. Scarrott, C.; MacDonald, A. A review of extreme value threshold estimation and uncertainty quantification. *REVSTAT-Stat. J.* **2012**, *10*, 33–60.
37. DuMouchel, W.H. Estimating the stable index α in order to measure tail thickness: A critique. *Ann. Stat.* **1983**, *11*, 1019–1031. [CrossRef]
38. Ferreira, A.; de Haan, L.; Peng, L. On optimising the estimation of high quantiles of a probability distribution. *Statistics* **2003**, *37*, 401–434. [CrossRef]
39. Loretan, M.; Phillips, P.C. Testing the covariance stationarity of heavy-tailed time series: An overview of the theory with applications to several financial datasets. *J. Empir. Financ.* **1994**, *1*, 211–248. [CrossRef]
40. Spangenberg, M.; Calmettes, V.; Julien, O.; Tourneret, J.Y.; Duchateau, G. Detection of variance changes and mean value jumps in measurement noise for multipath mitigation in urban navigation. *Navigation* **2010**, *57*, 35–52. [CrossRef]
41. L’Ecuyer, P. History of uniform random number generation. In *Proceedings of the 2017 Winter Simulation Conference (WSC)*, Las Vegas, NV, USA, 3–6 December 2017; pp. 202–230.
42. Maurya, M.R.; Rengaswamy, R.; Venkatasubramanian, V. Fault diagnosis using dynamic trend analysis: A review and recent developments. *Eng. Appl. Artif. Intell.* **2007**, *20*, 133–146. [CrossRef]
43. Strogatz, S.H. *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*; CRC Press: Boca Raton, FL, USA, 2018.
44. Rodriguez-Bermudez, G.; Garcia-Laencina, P.J. Analysis of EEG signals using nonlinear dynamics and chaos: A review. *Appl. Math. Inf. Sci.* **2015**, *9*, 2309.
45. Van Rossum, G.; Drake, F.L., Jr. *Python Tutorial*; Centrum voor Wiskunde en Informatica: Amsterdam, The Netherlands, 1995.
46. Oliphant, T.E. *A Guide to NumPy*; Trelgol Publishing USA, 2006; Volume 1. Available online: <https://ecs.wgtn.ac.nz/foswiki/pub/Support/ManualPagesAndDocumentation/numpybook.pdf> (accessed on 11 January 2020).
47. Jones, E.; Oliphant, T.; Peterson, P. *SciPy: Open Source Scientific Tools for Python*, 2001. Available online: <https://www.bibsonomy.org/bibtex/21b37d2cc741af879d7958f2f7c23c420/microcuts> (accessed on 11 January 2020).

48. Cejnek, M. Padasip—Open source library for adaptive signal processing in language Python. In *Studentská Tvůrčí činnost 2017*; Department of Instrumentation and Control Engineering: Prague, Czech Republic, 2017.
49. Hunter, J.D. Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.* **2007**, *9*, 90–95. [[CrossRef](#)]
50. Mackey, M.; Glass, L. Oscillation and chaos in physiological control systems. *Science* **1977**, *197*, 287–289. [[CrossRef](#)] [[PubMed](#)]
51. Zhang, J. Likelihood moment estimation for the generalized Pareto distribution. *Aust. N. Z. J. Stat.* **2007**, *49*, 69–77. [[CrossRef](#)]
52. Chen, H.; Cheng, W.; Zhao, J.; Zhao, X. Parameter estimation for generalized Pareto distribution by generalized probability weighted moment-equations. *Commun. Stat.-Simul. Comput.* **2017**, *46*, 7761–7776. [[CrossRef](#)]
53. Zhao, X.; Zhang, Z.; Cheng, W.; Zhang, P. A New Parameter Estimator for the Generalized Pareto Distribution under the Peaks over Threshold Framework. *Mathematics* **2019**, *7*, 406. [[CrossRef](#)]
54. Hanley, J.A.; McNeil, B.J. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology* **1982**, *143*, 29–36. [[CrossRef](#)]
55. Bradley, A.P. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognit.* **1997**, *30*, 1145–1159. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

ROC Analysis of Extreme Seeking Entropy for Trend Change Detection

Jan Vrba, Jan Mareš

University of Chemistry and Technology, Prague

Faculty of Chemical Engineering, Department of Computing and Control Engineering

Czech Republic

Email: jan.vrba@vscht.cz

Abstract—This paper is dedicated to the evaluation of the ROC curve of recently introduced Extreme Seeking Entropy algorithm. The ROC curve is evaluated for a trend change in the signal that contains additive Gaussian noise. The resulting ROC curve of the Extreme Seeking Entropy algorithm is compared with other adaptive novelty detection methods, namely Learning Entropy and Error and Learning Based Novelty Detection as those algorithms are also evaluating the adaptive weights increments. The ROC curves are evaluated for multiple noise variances and area under those ROC curves is estimated.

keywords—signal processing, adaptive systems, adaptive algorithms, novelty detection, trend change detection

I. INTRODUCTION

Novelty detection (ND) is still an important topic and adaptive methods are getting importance as more and more data are processed nowadays. In decades many different approaches have been developed [1], [2], [3]. However, every ND method is performing well for specific data sets, so there is no universal method at all and thus the new ND methods are still needed. Two main approaches can be recognized in the ND topic, statistical (i.e. [4]) and neural network approach. From this point of view, the Learning Entropy (LE) [5] and Error and Learning Based Novelty Detection (ELBND) [6] can be classified as neural network approaches. The recently introduced Extreme Seeking Entropy algorithm (ESE) [7] can be classified as a combination of both.

In this, article we are going to estimate a receiver operating characteristic (ROC) curves [8] of the LE, ELBND and ESE algorithms. The ROC curve is a useful tool to visualize the ability of binary classification systems to classify data as the classification threshold is changing as well as to compare the performance of classification systems [9]. The scenario of trend change detection, where data contains additive gaussian noise, is evaluated. The trend change detection is a problem that is related i.e. to fault diagnosis [10]. The ESE, LE and ELBND methods have been selected as all of them are evaluating the adaptive weight increments that are obtained via sample-by-sample learning algorithms and all of those methods are utilizing simple adaptive systems to detect novelty in the data.

II. ADAPTIVE SYSTEM AND LEARNING ALGORITHM SPECIFICATION

In the experiments, the simple higher order neural unit (HONU) adaptive model is used. The output of this model at a discrete time index k is given as

$$y(k) = w_1(k) \cdot x_1(k) + w_2(k) \cdot x_2(k) + w_3(k) \cdot x_1(k) \cdot x_2(k) \quad (1)$$

which is equivalent with form

$$y(k) = \mathbf{w}^T(k) \cdot \mathbf{x}(k) \quad (2)$$

where

$$\mathbf{w}^T(k) = [w_1(k), w_2(k), w_3(k)] \in R^3$$

is the vector of adaptive weights and

$$\mathbf{x}^T(k) = [x_1(k), x_2(k), x_1(k) \cdot x_2(k)] \in R^3$$

is the input vector. The vector of adaptive weights is updated with every new sample obtained, formally the update of the adaptive weight vector is given as

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \Delta\mathbf{w}(k), \quad (3)$$

where $\Delta\mathbf{w}(k)$ is the vector of weights increments. The initial setting of every adaptive weight is drawn from uniform distribution $U(-1, 1)$.

The generalized gradient descent (GNGD) algorithm was used for the experiments. The update of this adaptive algorithm is given as

$$\Delta\mathbf{w}(k) = \eta(k)e(k)\mathbf{x}(k) \quad (4)$$

where $e(k)$ is prediction error given as

$$e(k) = d(k) - y(k) \quad (5)$$

and

$$\eta(k) = \frac{\mu}{\|\mathbf{x}(k)\|_2^2 + \epsilon(k)}$$

$$\epsilon(k) = \epsilon(k-1) - \rho\mu \frac{e(k)e(k-1)\mathbf{x}^T(k)\mathbf{x}(k-1)}{(\|\mathbf{x}(k-1)\|_2^2 + \epsilon(k-1))^2}$$

where $d(k)$ is the output of the system, $\mu \in R$ is the learning rate, $\epsilon \in R$ is a compensation term, and ρ is the step size adaptation parameter, which should be chosen as $0 \leq \rho \leq 1$ to preserve the stability of the GNGD. The learning rate μ was set as $\mu = 0.5$ for all experiments.

III. NOVELTY DETECTION ALGORITHMS

A. Extreme Seeking Entropy Algorithm

The Extreme Seeking Entropy (ESE) algorithm [7] is utilizing the last n_s adaptive weights increments to detect novelty in the data. With every new adaptive weight increment obtained the peak-over-threshold (POT) method is proceed and if the increment is bigger than threshold ζ_i given by the POT method, the parameters of Generalized Pareto distribution are estimated and the probability of the weight increment is evaluated. Then ESE is calculated according to following formula (6).

$$ESE(|\Delta \mathbf{w}(k)|) = -\log \prod_{i=1}^n (1 - f_{cdf_i}(|\Delta w_i(k)|)) \quad (6)$$

where

$$f_{cdf_i}(|\Delta w_i(k)|) = \begin{cases} 0, & |\Delta w_i(k)| < \zeta_i \\ F_{(\xi_i, \mu_i, \sigma_i)}(|\Delta w_i(k)|), & |\Delta w_i(k)| \geq \zeta_i. \end{cases}$$

and $F_{(\xi_i, \mu_i, \sigma_i)}$ is cumulative distribution function of the generalized Pareto distribution with location parameter μ_i , scale parameter σ_i and shape parameter ξ_i and n is the number of adaptive weights.

Algorithm 1 Extreme Seeking Entropy Algorithm

- 1: set n_s and choose POT method
 - 2: initial estimation of the GPDs parameters ξ_i, μ_i, σ_i for each adaptable parameter
 - 3: **for** each new $d(k)$ **do**
 - 4: update adaptive model to get $\Delta \mathbf{w}(k)$
 - 5: proceed POT method
 - 6: **if** $|\Delta w_i(k)| > \zeta_i$ **then**
 - 7: update parameters of GPDs ξ_i, μ_i, σ_i
 - 8: **end if**
 - 9: compute ESE according to (6)
 - 10: **end for**
-

The POT method was chosen according to [11] as 10%, so the parameters of GPD are estimated from the 10% of the highest weight increments.

B. Error and Learning Based Novelty Detection Algorithm

Error and Learning Based Novelty Detection algorithm (ELBND) utilize the prediction error and adaptive weight increments [6]. The novelty in the obtained sample is given as follows.

$$ELBND(k) = \sum_{i=1}^n |e(k) \cdot \Delta w_i(k)| \quad (7)$$

Note that there are multiple approaches to estimate ELBND [6], for the experiments we used the sum version.

C. Learning Entropy Algorithm

Recently the new version of the Learning Entropy algorithm has been published [5]. The direct version avoids the need for multiple scales [12] and estimates the learning entropy of the obtained sample at discrete time index k as

$$LE(k) = \sum_{i=1}^n z(|\Delta w_i(k)|) \quad (8)$$

with special z - score that is given as

$$z(|\Delta w_i(k)|) = \frac{|\Delta w_i(k)| - \overline{|\Delta w_i^M(k-1)|}}{\sigma(|\Delta w_i^M(k-1)|)} \quad (9)$$

where $\overline{|\Delta w_i^M(k-1)|}$ is the mean of the last M adaptive i th weight increments and $\sigma(|\Delta w_i^M(k-1)|)$ is their standard deviation.

IV. EVALUATION OF THE ROC CURVE

In this section, the ESE algorithm is compared with the Learning Entropy algorithm and Error and Learning Based Novelty Detection algorithm. The comparison is evaluated in trend change detection case with the synthetic dataset. The structure of the experiment was chosen as it is in [7] to extend already published results.

A. Description of the Experiment

The output of the system d at the discrete time index k is given by the equation

$$d(k) = x_1(k) + x_2(x) + 0.01 \cdot k + v(k) \quad (10)$$

$$0 \leq k < 200$$

where $v(k)$ is additive Gaussian noise with zero mean and standard deviation σ_n and $x_1(k), x_2(k)$ are inputs with values drawn from the uniform distribution $U(-1, 1)$. At time $k = 200$ there is a step-change in the signal and the equation (10) changes to

$$d(k) = x_1(k) + x_2(x) + (0.01 + \delta) \cdot k + v(k) \quad (11)$$

$$200 \leq k \leq 399$$

where the value of parameter δ is drawn from the uniform distribution $U(-0.02, 0.02)$ and is constant for the $200 \leq k \leq 399$. The initial estimation of GPD parameters and LE values was done using 1200 samples that were obtained from the system given by equation (10). Note that for the ESE algorithm the value of was set to $n_s = 1200$.

B. Receiver Operating Characteristic

To construct the receiver operating characteristic (ROC) the dataset with the same number of positive and negative cases is desirable. Each performed experiment consists of 400 samples and totally 10 000 experiments were performed. To obtain a suitable dataset for ROC every experiment was re-sampled as follows.

$$ND_r(i) = \max\{ND(i \cdot 10), ND(i \cdot 10 + 1), ND(i \cdot 10 + 2), \dots, ND(i \cdot 10 + 9)\},$$

$$i = 0, 1, \dots, 39 \quad (12)$$

where $ND(k)$ represents the novelty measure in data at the discrete time index k (value of ESE , LE , $ELBND$). Every re-sampled data sequence was divided into two sets. Set P contain one positive sample $P = \{ND(20)\}$. Set N contain 39 negative samples, $N = \{ND(0), \dots, ND(19), ND(21), \dots, ND(39)\}$. For each experiment totally two samples that represents sets P and N were used to evaluate ROC, note that the sample from set N was drawn randomly. Totally there were 10 000 experiments performed for multiple values of noise variance σ_n . The values of noise variance were selected as follows $\sigma_n = \{0.1, 0.2, 0.5, 1.0, 2.0, 2.5\}$ and signal-to-noise ratio (SNR) is evaluated based from all experiments for each noise variance as

$$SNR = 10 \log_{10} \frac{\sigma_s}{\sigma_n} \quad (13)$$

where σ_s is the standard deviation of the output of the system. The True positive rate (TPR) correspond to hit rate and it is a ratio of true positive cases (TP) for a given threshold to the total number of positive cases.

$$TPR = \frac{TP}{10000} \quad (14)$$

False positive rate (FPR) is given as ratio of false positive cases (FP) for given threshold to total number of negative cases.

$$FPR = \frac{FP}{10000} \quad (15)$$

C. Results

The resulting ROC curves for various SNR are depicted in following Figures 1-6. The blue solid line depicts the results of the Extreme Seeking Entropy algorithm, the green dotted line Learning Entropy algorithm, the red dashed line Error and Learning Based Novelty Detection algorithm and black dash-dotted line random classifier. For each ROC curve the

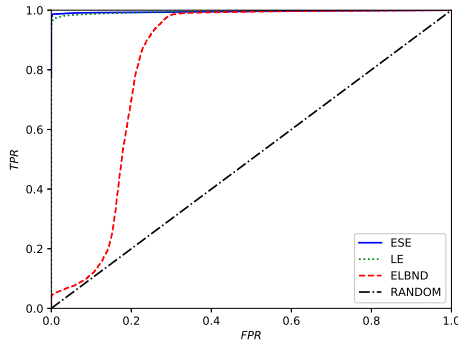


Figure 1. ROC curve for trend change detection of the signal containing additive Gaussian noise with variance $\sigma = 0.1$ and corresponding $SNR = 35.8$ dB. The acronym ESE is the Extreme Seeking Entropy, LE is the Learning Entropy, ELBND is the Error and Learning Based Novelty Detection method and RANDOM is the random classifier.

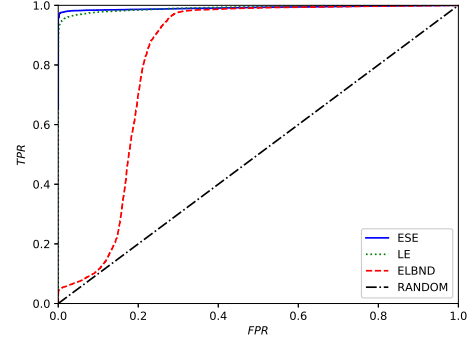


Figure 2. ROC curve for trend change detection of the signal containing additive Gaussian noise with variance $\sigma = 0.2$ and corresponding $SNR = 30.0$ dB. The acronym ESE is the Extreme Seeking Entropy, LE is the Learning Entropy, ELBND is the Error and Learning Based Novelty Detection method and RANDOM is the random classifier.

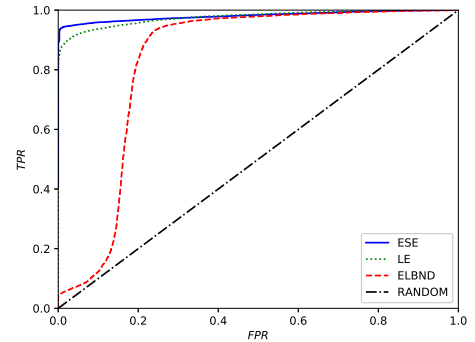


Figure 3. ROC curve for trend change detection of the signal containing additive Gaussian noise with variance $\sigma = 0.5$ and corresponding $SNR = 21.7$ dB. The acronym ESE is the Extreme Seeking Entropy, LE is the Learning Entropy, ELBND is the Error and Learning Based Novelty Detection method and RANDOM is the random classifier.

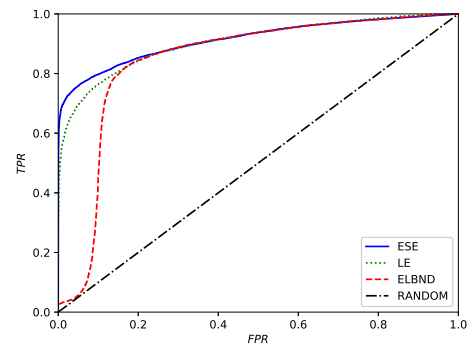


Figure 4. ROC curve for trend change detection of the signal containing additive Gaussian noise with variance $\sigma = 1.0$ and corresponding $SNR = 16.2$ dB. The acronym ESE is the Extreme Seeking Entropy, LE is the Learning Entropy, ELBND is the Error and Learning Based Novelty Detection method and RANDOM is the random classifier.

area under the curve ($AUROC$) was estimated using

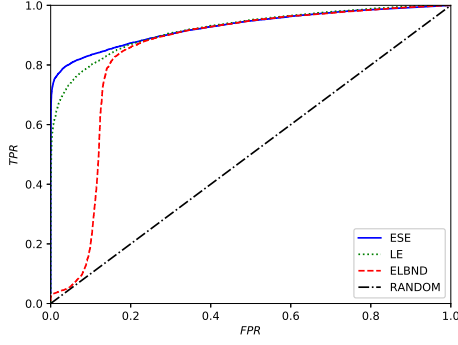


Figure 5. ROC curve for trend change detection of signal containing additive Gaussian noise with variance $\sigma = 2.0$ and corresponding $SNR = 10.88$ dB. The acronym ESE is the Extreme Seeking Entropy, LE is the Learning Entropy, ELBND is the Error and Learning Based Novelty Detection method and RANDOM is the random classifier.

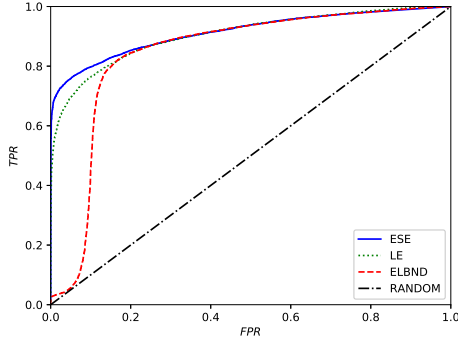


Figure 6. ROC curve for trend change detection of the signal containing additive gaussian noise with variance $\sigma = 2.5$ and corresponding $SNR = 9.2$ dB. The acronym ESE is the Extreme Seeking Entropy, LE is the Learning Entropy, ELBND is the Error and Learning Based Novelty Detection method and RANDOM is the random classifier.

trapezoidal rule:

$$AUROC = \int_0^1 TRP(FPR) \approx \sum_{j=1}^{19999} \frac{TPR(FPR(j)) + TPR(FPR(j+1))}{2} \cdot (FPR(j+1) - FPR(j)) \quad (16)$$

and resulting $AUROC$ s are in following table I. For all six experiments, the detection rate was evaluated. The successful detection is considered if the maximum in the ND score corresponds with the positive sample. The results of detection rates are depicted in the following table II.

V. CONCLUSION

This paper is presenting the ROC curves for trend change detection problem with artificial data. The ROC curves and area under ROC curve are evaluated for various values of SNR as the signal contains additive Gaussian noise. The comparison of Extreme seeking entropy algorithm, Learning entropy algorithm and Error and Learning Based Novelty detection algorithm

TABLE I
 $AUROC$ FOR TREND CHANGE DETECTION

		AUROC		
σ_n	SNR [dB]	ESE	LE	ELBND
0.1	35.8	0.9954	0.9952	0.8234
0.2	30.0	0.9920	0.9912	0.8299
0.5	21.7	0.9816	0.9777	0.8288
1.0	16.2	0.9576	0.9496	0.8263
2.0	10.8	0.9286	0.9214	0.8397
2.5	9.2	0.9134	0.9056	0.8446

TABLE II
DETECTION RATES FOR TREND CHANGE DETECTION

		Detection rate		
σ_n	SNR [dB]	ESE	LE	ELBND
0.1	35.8	98.88	98.92	60.00
0.2	30.0	98.14	98.03	59.61
0.5	21.7	95.18	95.08	59.65
1.0	16.2	90.42	89.96	57.67
2.0	10.8	81.27	78.51	57.69
2.5	9.2	75.86	71.56	57.16

is shown. From the obtained results it seems that the ESE and the LE algorithms slightly outperform the ELBND algorithm in the presented experiments. The drawback of using ESE is its higher computational complexity.

ACKNOWLEDGMENT

The authors would like to thank Matous Cejnek for developing PADASIP (Python Adaptive Signal Processing library). This work was supported from the grant of Specific university research – grant No. A1_FCHI_2020_002.

REFERENCES

- [1] M. Markou and S. Singh, “Novelty detection: A review - part 2: Neural network based approaches,” *Signal Processing*, vol. 83, pp. 2499–2521, 2003.
- [2] M. Markou and S. Singh, “Novelty detection: a review—part 1: statistical approaches,” *Signal Processing*, vol. 83, no. 12, pp. 2481 – 2497, 2003.
- [3] A. Zimek and P. Filzmoser, “There and back again: Outlier detection between statistical reasoning and data mining algorithms,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, Aug. 2018.
- [4] M. Desforages, P. Jacob, and J. Cooper, “Applications of probability density estimation to the detection of abnormal conditions in engineering,” *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 212, no. 8, pp. 687–703, 1998.
- [5] I. Bukovsky, W. Kinsner, and N. Homma, “Learning entropy as a learning-based information concept,” *Entropy*, vol. 21, p. 166, feb 2019.
- [6] M. Cejnek and I. Bukovsky, “Concept drift robust adaptive novelty detection for data streams,” *Neurocomputing*, vol. 309, pp. 46–53, 2018.
- [7] J. Vrba and J. Mareš, “Introduction to extreme seeking entropy,” *Entropy*, vol. 22, p. 93, Jan. 2020.
- [8] J. P. Egan, *Signal detection theory and ROC-analysis*. Academic press, 1975.
- [9] T. Fawcett, “An introduction to roc analysis,” *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.
- [10] M. R. Maurya, R. Rengaswamy, and V. Venkatasubramanian, “Fault diagnosis using dynamic trend analysis: A review and recent developments,” *Engineering Applications of Artificial Intelligence*, vol. 20, pp. 133–146, Mar. 2007.
- [11] W. H. DuMouchel, “Estimating the stable index α in order to measure tail thickness: A critique,” *The Annals of Statistics*, vol. 11, pp. 1019–1031, Dec. 1983.
- [12] I. Bukovsky, “Learning entropy: Multiscale measure for incremental learning,” *Entropy*, vol. 15, pp. 4159–4187, Sept. 2013.

Computational Performance of the Parameters Estimation in Extreme Seeking Entropy Algorithm

Jan Vrba, Jan Mareš

University of Chemistry and Technology, Prague

Faculty of Chemical Engineering, Department of Computing and Control Engineering

Czech Republic

Email: jan.vrba@vscht.cz

Abstract—This paper is dedicated to the evaluation of the computational time performance of the algorithms that estimate the parameters of the generalized Pareto distribution, namely Method of Moments, Maximum likelihood estimator and Quasi-maximum likelihood algorithms. The generalized Pareto distribution is utilized by the Extreme Seeking Entropy algorithm to detect novelty in data. The algorithm is evaluating the weight increments of the simple adaptive filter that are obtained via incrementally learning algorithm. The computational time performance is examined in the experiment with the detection of step-change parameters of the signal generator. Its output contains also additive Gaussian noise.

keywords—signal processing, adaptive systems, adaptive algorithms, novelty detection, generalized Pareto distribution

I. INTRODUCTION

The real-time novelty detection is the area that is associated not only with fault detection and diagnostics [1] in industrial processes but also with i.e. detection of anomalies in data streams as those anomalies may be associated with network intrusion or fraud attempts [2], or may be utilized in surveillance security systems [3]. Another wide area of real-time applications is mobile robotics where the computational power is usually limited [4], [5]. The key feature for each novelty detection method that is useful in real-time applications is outstanding computational time performance.

This paper is devoted to comparison of the computational time performance of three different methods for the generalized Pareto distribution [6] (GPD) parameters estimation. Compared methods are a maximum likelihood estimator (ML), method of moments (MOM) and quasi-maximum likelihood estimator (QML). The computational time performance of those methods is evaluated in the experiment of signal generator parameters step-change detection via adaptive novelty detection algorithm Extreme Seeking Entropy (ESE).

II. ADAPTIVE SYSTEM AND ALGORITHM SPECIFICATION

As the aim of the article is the real-time performance of algorithms, that estimate the GPD parameters and not the novelty detection in a complex processes, the simple linear finite impulse response filter is used. The output of the filter y at a discrete time index k is described by the following equation (1) so that

$$y(k) = w_1(k) \cdot x_1(k) + w_2(k) \cdot x_2(k) + w_3(k) \cdot x_3(k) \quad (1)$$

which is equivalent to vector form

$$y(k) = \mathbf{w}^T(k) \cdot \mathbf{x}(k) \quad (2)$$

where

$$\mathbf{w}^T(k) = [w_1(k), w_2(k), w_3(k)] \in R^3$$

is the vector of filters adaptive weights and

$$\mathbf{x}^T(k) = [x_1(k), x_2(k), x_3(k)] \in R^3$$

is the input vector. The adaptive weights are updated using the NLMS algorithm [7] with every new sample obtained. The new value of adaptive weights is given as

$$\Delta \mathbf{w}(k) = \frac{\kappa \cdot \mathbf{w}(k) \cdot e(k)}{\varepsilon + \mathbf{x}(k) \cdot \mathbf{x}^T(k)}, \quad (3)$$

where ε is used to avoid division by zero in case of zero input vector, $\kappa \in R$ is the learning rate and $e \in R$ is the output error defined as

$$e(k) = d(k) - y(k), \quad (4)$$

where $d(k) \in R$ is the measured value of the output and k is the discrete time index. For the experiment the value of the learning rate κ was set as $\kappa = 0.8$

III. EXTREME SEEKING ENTROPY ALGORITHM

The ESE algorithm [8] estimates the novelty in data via the probability of adaptive weights increments. Those increments are obtained via incrementally learning algorithm and the probability is estimated via generalized Pareto distribution. The parameters of GPD are obtained using the last n_s adaptive weight increments

on which is applied the peak-over-threshold (POT) method so for parameters estimation only n biggest weight increments are used. The POT method was chosen according to [9] as 10%, so the parameters (denoted n) of GPD are estimated from the 10% of the highest weight increments.

$$n = \lceil 0.1 \cdot n_s \rceil \quad (5)$$

The novelty score is calculated according to following formula (6).

$$ESE(|\Delta \mathbf{w}(k)|) = -\log \prod_{i=1}^{n_f} (1 - f_{cdf_i}(|\Delta w_i(k)|)) \quad (6)$$

where

$$f_{cdf_i}(|\Delta w_i(k)|) = \begin{cases} 0, & |\Delta w_i(k)| < \zeta_i \\ F_{(\gamma_i, \mu_i, \sigma_i)}(|\Delta w_i(k)|), & |\Delta w_i(k)| \geq \zeta_i. \end{cases}$$

and $F_{(\gamma_i, \mu_i, \sigma_i)}$ is cumulative distribution function of the generalized Pareto distribution with location parameter μ_i , scale parameter σ_i and shape parameter γ_i and n_f is the number of adaptive weights and ζ_i correspond to the minimum of the highest adaptive weight increments that were obtained after proceeding the peak-over-threshold method.

Algorithm 1 Extreme Seeking Entropy Algorithm

- 1: set n_s and choose POT method
 - 2: initial estimation of the GPDs parameters $\gamma_i, \mu_i, \sigma_i$ for each adaptable parameter
 - 3: **for** each new $d(k)$ **do**
 - 4: update adaptive model to get $\Delta \mathbf{w}(k)$
 - 5: proceed POT method
 - 6: **if** $|\Delta w_i(k)| > \zeta_i$ **then**
 - 7: update parameters of GPDs $\gamma_i, \mu_i, \sigma_i$
 - 8: **end if**
 - 9: compute ESE according to (6)
 - 10: **end for**
-

A. Estimation of Generalized Pareto Distribution Parameters

Assume the three parameters form of GPD so that probability density function is defined as follows.

$$f_{(\gamma, \mu, \sigma)}(w) = \begin{cases} \frac{1}{\sigma} \left(1 + \frac{\gamma(w-\mu)}{\sigma} \right)^{-\frac{1}{\gamma}-1} & \text{for } \gamma \neq 0, \\ \exp \left(-\frac{w-\mu}{\sigma} \right) & \text{for } \gamma = 0. \end{cases} \quad (7)$$

Note that parameter $\mu \in (-\infty, +\infty)$ is a location parameter, $\sigma \in (0, \infty)$ is the scaling parameter, and $\gamma \in (-\infty, \infty)$ is a shape parameter. Then the cumulative distribution function then is given as

$$F_{(\gamma, \mu, \sigma)}(w) = \begin{cases} 1 - \left(1 + \frac{\gamma(w-\mu)}{\sigma} \right)^{-\frac{1}{\gamma}} & \text{for } \gamma \neq 0, \\ 1 - \exp \left(-\frac{w-\mu}{\sigma} \right) & \text{for } \gamma = 0. \end{cases} \quad (8)$$

A ML estimator of GPD parameters (σ, γ) has to be found numerically maximizing the log-likelihood function given as

$$\log \mathcal{L}(\sigma, \gamma | w_1, \dots, w_n) = -n \log \sigma + \frac{1-\gamma}{\gamma} \sum_n \log \left(1 - \frac{\gamma}{\sigma} (w_i - \mu) \right) \quad (9)$$

with $\forall w_i \geq \mu$ (note that in this section the index of adaptive weight w is associated with number of sample that is obtained after POT method processing, not with the number of adaptive weights). The solving problem (9) is difficult, as the optimization method can converge to a sub-optimal solution and the explicit expressions of the ML estimator do not exist. Another issue that is related to the real-time application is that it is not possible to guarantee the maximum computational time dedicated to solving the problem.

A method of moments estimator of parameters (σ, γ) [10] is feasible if the parameter of the GPD $\gamma < 0.5$. Then the first and second raw moments are defined and the parameters of the GPD (σ, γ) are computed as follows

$$\hat{\sigma} = \frac{1}{2} \bar{w} \left(\frac{\bar{w}^2}{s^2} + 1 \right) \quad (10)$$

$$\hat{\gamma} = -\frac{1}{2} \left(\frac{\bar{w}^2}{s^2} - 1 \right) \quad (11)$$

where \bar{w} is sample mean and s^2 is sample variance.

The quasi-ML method [11] of estimation GPD parameters (σ, γ) is having following steps, assuming that (w_1, \dots, w_n) is ordered sequence such that $w_1 \leq w_2 \leq \dots \leq w_n$.

- 1) Compute

$$\hat{\gamma} = \frac{1}{n-1} \sum_{i=1}^{n-1} \ln \left(1 - \frac{w_i}{\max(w_1, \dots, w_n)} \right) \quad (12)$$

$$Z = 1 - \frac{n^{-1} \sum_{i=1}^n w_i^2}{2\bar{w}^2} \quad (13)$$

- 2) If $\hat{\gamma}$ and $Z < 0.2$ compute standard MLEs for $\hat{\gamma}$ and σ .
- 3) Otherwise estimate $\hat{\gamma}$ according to equation 12 and estimate $\hat{\sigma}$ as follows

$$\hat{\sigma} = \hat{\gamma} \cdot \max(w_1, \dots, w_n). \quad (14)$$

The accuracy of mentioned methods is not aim of this article as it was already evaluated in publications [11], [12], [13].

IV. EXPERIMENTAL RESULTS

The computational time performance was evaluated for the problem of the step-change parameters of the signal generator. Assume that the output of the signal generator $d(k)$ at the discrete time index k is given as

$$d(k) = x_1(k) + x_2(k) + x_3(k) + v(k) \quad (15)$$

for all $1 \leq k \leq 200$. Note that the $v(k)$ represents the additive gaussian noise with zero mean and standard

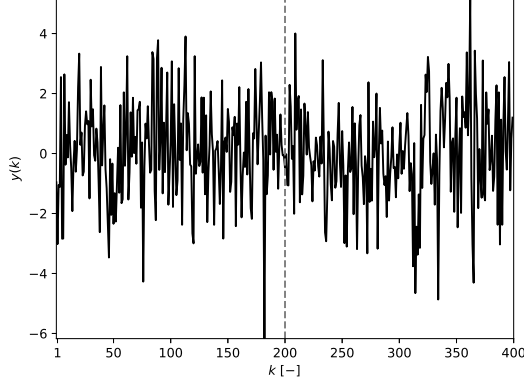


Figure 1. Output of the adaptive filter during the experiment. The step-change of the signal generator parameters at the discrete time index $k = 200$ is highlighted by grey dashed line.

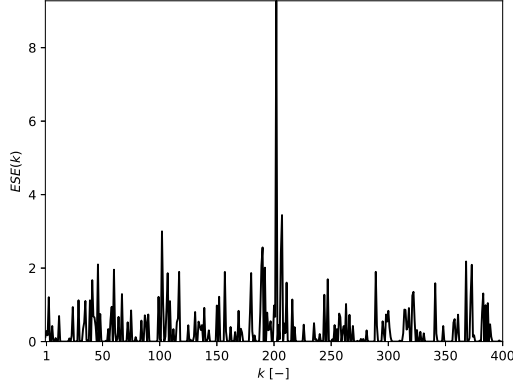


Figure 2. ESE value during the experiment. Note that the peak of ESE corresponds to the step-change of the signal generator.

deviation $\sigma_{noise} = 0.1$. At the discrete time index $k = 201$ the output of the signal generator changes to

$$d(k) = 0.7 \cdot x_1(k) + 1.2 \cdot x_1(k) + 1.1 \cdot x_1(k) + v(k) \quad (16)$$

for $201 \leq k \leq 400$. The value for each input of the signal generator is drawn from the standard normal distribution $\mathcal{N}(0, 1)$ so the i th input $x_i(k) \sim \mathcal{N}(0, 1)$ for $1 \leq k \leq 400$. Note, that the change of the signal generator parameters does not change the mean of the output $d(k)$. The output of the system generator is depicted in the following figure 1. The value of ESE is in the figure 2. The values of the GPD parameters (μ, σ, γ) during the experiment that were estimated by the ML are in the figures 3, 4 and 5.

All the experiments were performed on PC using Intel(R) Core(TM) i5-7400 4 cores CPU running at 3001 MHz and 32 GB RAM. The running operating system was Windows 10 Pro 64 bit version 10.0.18362 and code was written in Python 3.6.1 using Numpy 1.17.0 and Scipy 1.4.1 libraries. The average computational time \bar{t} of the estimation of all three GPDs parameters (the number of GPDs corresponds to the number of adaptive system parameters) and corresponding standard deviations σ_t are in the following table IV.

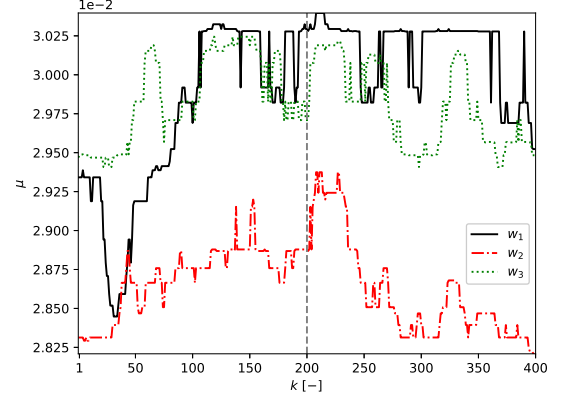


Figure 3. The values of the GPD location parameter μ for all three adaptive weights (w_1, w_2, w_3) during the experiment. The dashed line at the discrete time index k highlights the step-change of the signal generator parameters.

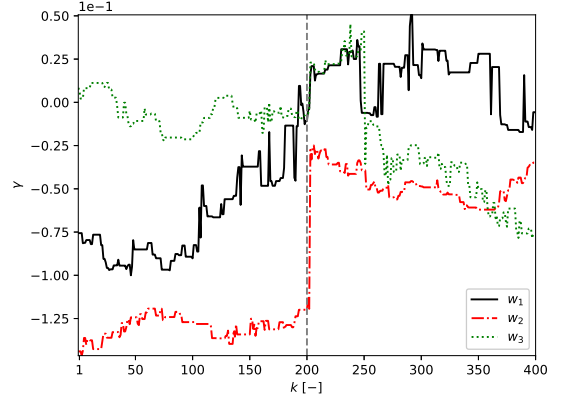


Figure 4. The values of the GPD shape parameter γ for all three adaptive weights (w_1, w_2, w_3) during the experiment. The dashed line at the discrete time index k highlights the step-change of the signal generator parameters.

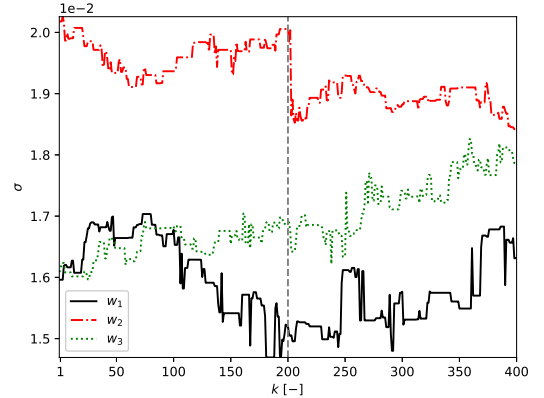


Figure 5. The values of the GPD scaling parameter σ for all three adaptive weights (w_1, w_2, w_3) during the experiment. The dashed line at the discrete time index k highlights the step-change of the signal generator parameters.

TABLE I
TABLE OF AVERAGE COMPUTATIONAL TIMES AND
CORRESPONDING STANDARD DEVIATIONS FOR SELECTED
METHODS

	Method	\bar{t} [ms]	σ_t [ms]
w_1	ML	26.198	3.396
	QML	0.354	0.478
	MOM	0.076	0.264
w_2	ML	26.718	2.302
	QML	0.337	0.471
	MOM	0.064	0.244
w_3	ML	24.982	1.964
	QML	0.395	0.489
	MOM	0.060	0.238

V. CONCLUSIONS

The average computational time and its standard deviations of three different methods (ML, QML, MOM) for estimating the GPD parameters were evaluated in this article. Estimation of those parameters is crucial for successful adaptive novelty detection via the Extreme Seeking Entropy algorithm. The evaluation was done for step-change of signal generator parameters. The output of the signal generator contains additive Gaussian noise with zero mean.

The fastest method from evaluated methods is MOM. The drawback of this method is that the usage in real-time applications is limited by the strict restriction on values of the GPD parameters. If the parameters do not meet the restriction, the estimation is inaccurate and the results of the ESE algorithm may be misleading. In general, it is hard to guarantee, that the monitored process is going to satisfy those restrictions.

The least computational efficient method is the ML, which takes on the order four hundred times more computational time than MOM in the tested scenario. On the other side, it provides estimation for a wider range of GPD parameters values. The higher value of computational time is natural for ML method as the optimization of the max-likelihood function is an iterative process where the number of iterations can not be guaranteed. However, the apriori information about the value parameters may speed-up the convergence.

The QML method seems like a good compromise between MOM and ML as it is fast enough and in the first step tests the need for using the ML method. If there is no need of using the ML method, the parameters of GPD are computed directly without the need of using the optimization method. It takes only about five times more computational time than MOM on average and provides the estimation for wider ranges of the GPD parameters. Note that even if the ML estimation has to be done, the computational time between the QML using ML and QML with direct computation of the parameters is not significant.

Based on the performed experiments it seems, that the usage of the ESE algorithm seems promising in real-time application if the computational power is sufficient, and if the data from the monitored process are available at a suitable rate.

ACKNOWLEDGMENT

The authors would like to thank Matous Cejnek for developing PADASIP (Python Adaptive Signal Processing library). This work was supported from the grant of Specific university research – grant No. A1_FCHI_2020_002.

REFERENCES

- [1] J. Gertler, *Fault detection and diagnosis in engineering systems*. CRC press, 1998.
- [2] K. Yu, W. Shi, N. Santoro, and X. Ma, “Real-time outlier detection over streaming data,” in *2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, pp. 125–132, IEEE, 2019.
- [3] R. Ramezani, P. Angelov, and X. Zhou, “A fast approach to novelty detection in video streams using recursive density estimation,” in *2008 4th International IEEE Conference Intelligent Systems*, IEEE, Sept. 2008.
- [4] S. Marsland, U. Nehmzow, and J. Shapiro, “On-line novelty detection for autonomous mobile robots,” *Robotics and Autonomous Systems*, vol. 51, pp. 191–206, May 2005.
- [5] U. Nehmzow, Y. Gatsoulis, E. Kerr, J. Condell, N. Siddique, and T. M. McGuinness, “Novelty detection as an intrinsic motivation for cumulative learning robots,” in *Intrinsically Motivated Learning in Natural and Artificial Systems*, pp. 185–207, Springer, 2013.
- [6] H. T. Davis and M. L. Feldstein, “The generalized pareto law as a model for progressively censored survival data,” *Biometrika*, vol. 66, no. 2, pp. 299–306, 1979.
- [7] S. S. Haykin, *Adaptive filter theory*. Pearson Education India, 2005.
- [8] J. Vrba and J. Mareš, “Introduction to extreme seeking entropy,” *Entropy*, vol. 22, p. 93, Jan. 2020.
- [9] W. H. DuMouchel, “Estimating the stable index α in order to measure tail thickness: A critique,” *The Annals of Statistics*, vol. 11, pp. 1019–1031, Dec. 1983.
- [10] J. R. Hosking and J. R. Wallis, “Parameter and quantile estimation for the generalized pareto distribution,” *Technometrics*, vol. 29, pp. 339–349, Aug. 1987.
- [11] A. Luceño, “Fitting the generalized pareto distribution to data using maximum goodness-of-fit estimators,” *Computational Statistics & Data Analysis*, vol. 51, no. 2, pp. 904–917, 2006.
- [12] V. Brazauskas and A. Kleefeld, “Robust and efficient fitting of the generalized pareto distribution with actuarial applications in view,” *Insurance: Mathematics and Economics*, vol. 45, no. 3, pp. 424–435, 2009.
- [13] V. Singh and H. Guo, “Parameter estimation for 3-parameter generalized pareto distribution by the principle of maximum entropy (pome),” *Hydrological sciences journal*, vol. 40, no. 2, pp. 165–181, 1995.

B Případová studie použití algoritmu Learning Entropy a adaptivního fuzzy filtru pro detekci změn stavů bioproduktu

Cílem této studie je ověřit, že algoritmus LE, který byl doposud publikován s adaptivními filtry typu LNU a HONU, je možné použít i pro jiné typy adaptivních filtrů. Pro provedenou studii tak byl vybrán adaptivní fuzzy filtr (viz kapitola 2.1.4).

B.1 Popis bioproduktu a specifikace problému

Podle [149] je pro fermentační procesy, které probíhají v dávkovém režimu je podstatné, aby probíhalo správně dávkované živiny substrátem. Pro tyto procesy je specifické, že se při nadměrných koncentracích stává substrát pro mikroorganismy toxickým a může dojít k tzv. přežití a tím i zahubení těchto mikroorganismů. Naopak, v důsledku nedostatečného zábovení živinami může dojít k odumření kultivovaného organismu. Z tohoto pohledu se je tedy důležité v závislosti na koncentraci substrátu a stavu populace mikroorganismů měnit i strategii pro řízení procesu kultivace. Historicky byl stav bioproduktu klasifikován expertem, přičemž vyhodnocení bylo poměrně časově náročné a nebylo neobvyklé, že různí experti docházeli k rozdílným závěrům. Protože výnos fermentačního procesu je zásadním způsobem ovlivněn správnou klasifikací stavu ve kterém se právě nachází, bylo by vhodné klasifikaci automatizovat a pokud možno zvýšit její přesnost. Tomuto problému je právě věnována publikace [149], která řeší problém automatické klasifikace stavů bioproduktu kultivace bakterie *Pseudomonas putida* KT2442. V této publikaci je navržen komplexní algoritmus pro online klasifikaci stavů bioproduktu. Autoři zde rozlišují celkem tři stavy bioproduktu kultivace *Pseudomonas putida* KT2442, konkrétně:

1. normální živiny
2. přežití
3. nedoživiny

Navržený algoritmus vyhodnocuje přísun vstupujících živin (F_m), respektive substrátu, a změny a trendu rozpuštěného kyslíku (DO), který je produkován bakteriemi *Pseudomonas putida* a pomocí hřebenové regrese (v literatuře se vyskytuje také pod názvem Tichonova regularizace) je určován vývoj populace bakterií respektive stav bioproduktu. Vzhledem k tomu, že modely vývoje populace pro jednotlivé stavy jsou různé, mohlo by v průběhu experimentu dojít i k podstatným změnám v adaptivním modelu v okamžicích změn stavů kultivace. Tyto změny se mohli projevit neobvykle velkými přírůstky adaptivních parametrů.

Protože algoritmus Learning Entropy využívá přírůstek adaptivních parametrů, mohl by být vhodným nástrojem pro detekci změn stavů bioproduktu. Předpokládáme, že tedy existuje souvislost mezi změnami stavu bioproduktu a nárůstem Learning Entropy. Přestože může být

proces kultivace bakterií *Pseudomonas Putidas* modelován různými a různě složitými modely, pro využití algoritmu Learning Entropy se jeví výhodné použít jednoduché prediktory nebo sledovače. Dosud publikované články využívali pro algoritmus LE pouze FIR filtry, případně Volterrovy filtry, které mají adaptivní parametry v lineární závislosti. V tomto experimentu je použit adaptivní fuzzy filtr, jehož struktura je specifikována v kapitole 2.1.4 a k jehož adaptaci byl použit algoritmus, který je uveden v kapitole 2.2.4.

Použitý adaptivní fuzzy filtr má 9 pravidel ($M = 9$), jehož l -té pravidlo je ve tvaru

$$IF\ do(k-1)\ is\ A_1^l\ AND\ do(k-6)\ is\ A_2^l\ AND\ do(k-19)\ is\ A_3^l\ THEN\ do(k)\ is\ B^l \quad (1)$$

kde A_i^l je množina ve vstupním prostoru $U \subset R^3$ a B^l je fuzzy množina ve výstupním prostoru $V \subset R$. V uvedeném pravidle jsou $do(k-1)$, $do(k-6)$, $do(k-19)$ a $do(k)$ lingvistické proměnné, které vyjadřují koncentraci rozpuštěného kyslíku do v diskretních časových okamžicích k , $k-1$, $k-6$ respektive $k-19$, kde k je diskretní časový index. Vzhledem k tomu, že uvedený adaptivní fuzzy filtr používá Gaussovske funkce příslušnosti (viz kapitola 2.1.4 je zobrazení popisující jeho výstup ve tvaru

$$\hat{y}(\mathbf{x}(k)) = \frac{\sum_{j=1}^9 \bar{b}^j \left[\prod_{i=1}^3 \exp\left(-\left(\frac{x_i - \bar{x}_i^j}{\sigma_i^j}\right)^2\right) \right]}{\sum_{j=1}^9 \left[\prod_{i=1}^3 \exp\left(-\left(\frac{x_i - \bar{x}_i^j}{\sigma_i^j}\right)^2\right) \right]} \quad (2)$$

kde vektor $\mathbf{x}(k)$ je

$$\mathbf{x}(k) = [do(k-1), do(k-6), do(k-19)]. \quad (3)$$

Protože hodnota koncentrace rozpuštěného kyslíku je uváděna v procentech, platí pro všechna $x_i \in \langle 0; 100 \rangle$. K adaptování výše uvedeného filtru byl použit algoritmus gradient descent (viz kapitola 2.2.4. Maximální počet epoch byl stanoven na $q_{max} = 100$ a požadovaná chyba predikce mezi výstupem adaptivního filtru a naměřenými daty na $\epsilon = 0,001$. Rychlost učení byla během experimentů nastavena na $\mu = 1$.

Pro vyhodnocení novosti byla použita přímá verze algoritmu LE, takže

$$E(k) = \max\{0, \sum_{j=1}^9 z(|\Delta \bar{b}^j|) - \beta\} \quad (4)$$

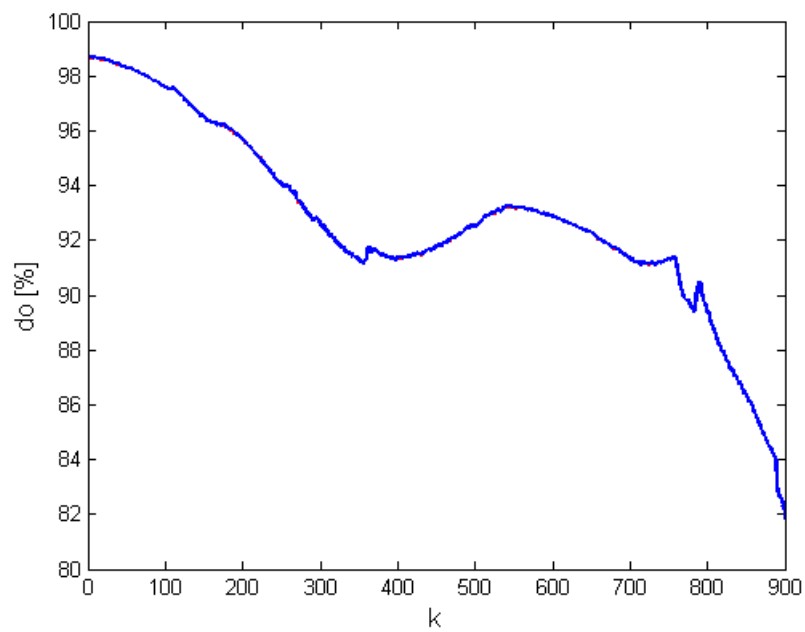
kde funkce z je dána rovnicí (XY) a β je citlivostní parametr. Pro vyhodnocení novosti jsou tedy použity změny polohy středů množin ve výstupním prostoru, nikoliv změny parametrů fuzzy množin ve vstupním prostoru.

B.2 Experiment a zhodnocení

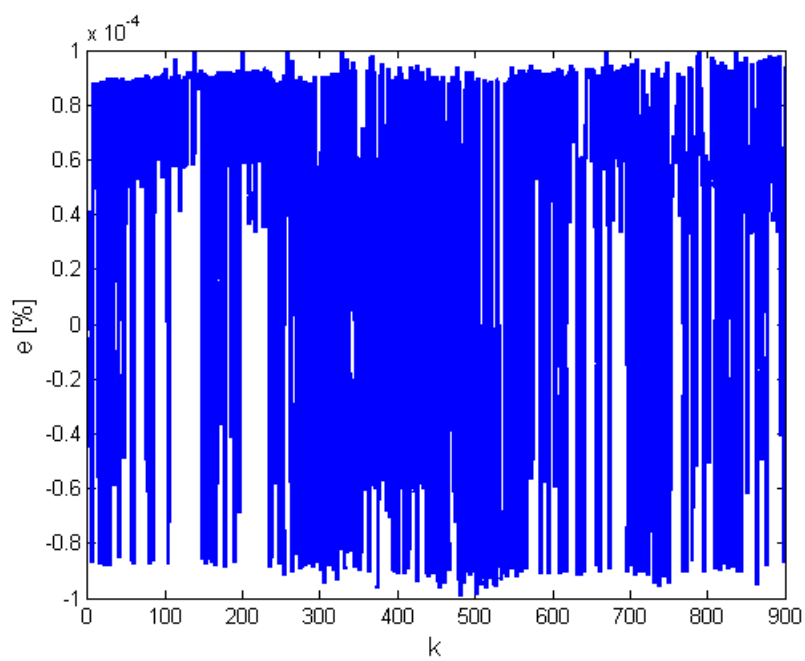
Experiment s použitím algoritmu LE a adaptivního fuzzy filtru byl uskutečněn na datech z kultivace bakterie *Pseudomonas Putida*, který byl uskutečněn na Ústavu počítačové a řídicí techniky VŠCHT Praha. Celkem byly zpracovávány hodnoty ze dvou kultivací. Přestože bylo během experimentu měřena sada různých veličin (např. teplota, pH, atd.), osvědčil se pro použití LE signál rozpustěného kyslíku do [%]. Během experimentu bylo použito vzorkování $T = 1 \text{ min}$, což je vzhledem k rychlosti celého procesu dostatečně rychlé vzorkování. Počáteční nastavení parametrů adaptivního fuzzy filtru bylo provedeno tak, jak je popsáno v kapitole 2.2.4. Podstatný vliv na výsledek detekce změn stavu bioprocesu měla volba délky okna pro vyhodnocení změn adaptabilních parametrů filtru M_{ND} . Na základě experimentů s různými délkami byla nakonec zvolena délka okna $M_{ND} = 20$.

Na následujících obrázcích je znázorněn průběh signálu do během první kultivace (viz obrázek 1), chyba predikce (viz obrázek 2) a odpovídající hodnoty LE společně se stavy bioprocesu (viz obrázek 5). Význam stavů bioprocesu znázorněných na obrázku 2 respektive obrázku 6 jsou: 1 - nedoživení, 2 - živení, 3 - přeživení. Aby byli detekovány všechny změny stavu bioprocusů, byla stanovena hodnota parametru $\beta = 2,58$ (viz rovnice 4).

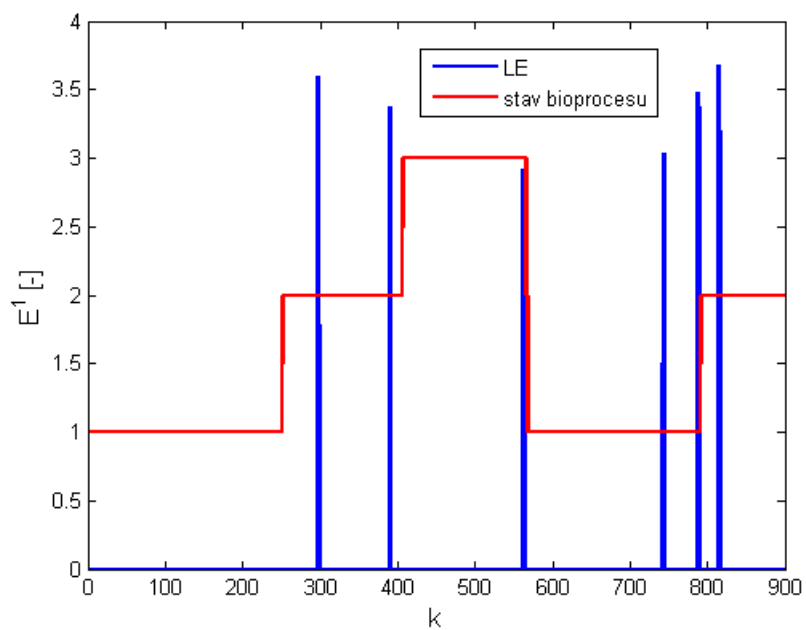
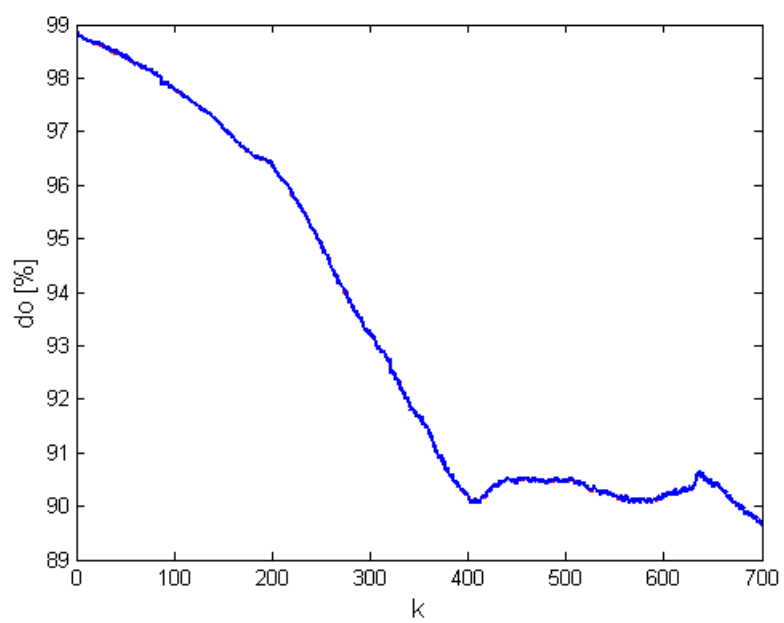
Data z druhé kultivace byla použita k ověření správného nastavení parametru β . Obrázek 4 zobrazuje průběh signálu do během druhého experimentu. Chyba predikce adaptivního fuzzy filtru je zobrazena na obrázku 3. Obrázek 6 zobrazuje stavy bioprocesu během kultivace a odpovídající hodnoty LE . S tímto nastavením parametru β se podařilo detekovat pouze tři změny bioprocesu. Nicméně pro jiné hodnoty délky okna M_{ND} a hodnoty parametru β se tyto změny detekovat podařilo. Je tedy zřejmé, že pro praktické použití je správná volba obou parametrů zásadní. Vzhledem k malému množství dat a časové náročnosti kultivace nebylo možné použít nějakou validační metodu.

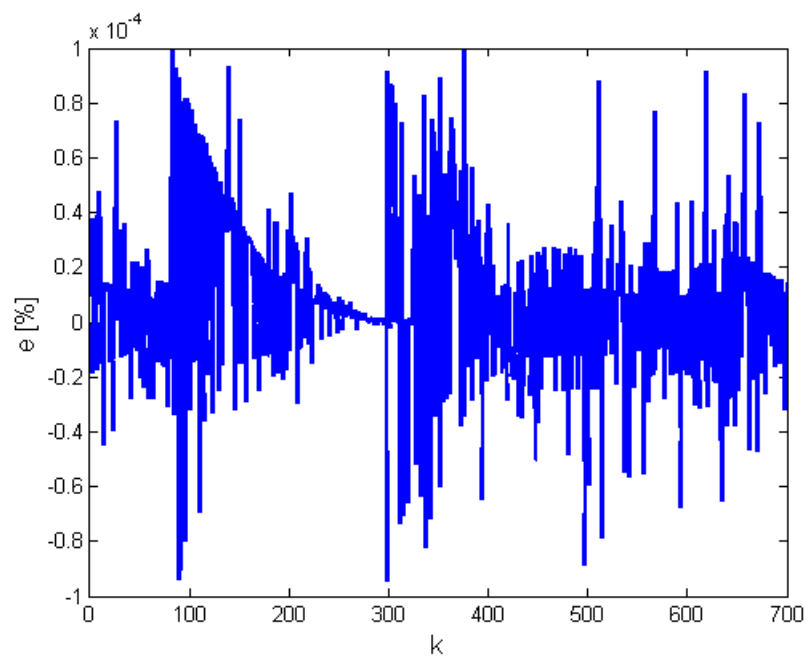


Obrázek 1: Průběh signálu do během první kultivace

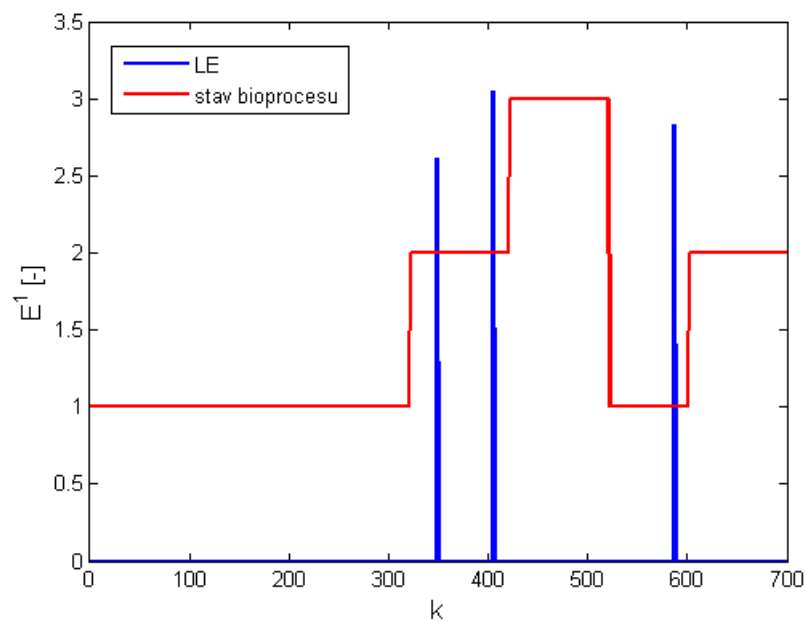


Obrázek 2: Chyba predikce e během první kultivace

Obrázek 3: Stav bioprocessu a hodnota LE první kultivaceObrázek 4: Průběh signálu do během druhé kultivace



Obrázek 5: Chyba predikce e během druhé kultivace



Obrázek 6: Stav bioprocesu a hodnota LE během druhé kultivace

C Vyhodnocení přesnosti detekce změny trendu a skokové změny parametrů generátoru signálu

SNR [dB]	ESE [%]	ELBND [%]	LE [%]	Err [%]
41.63	99.0	68.1	98.9	54.8
35.62	98.4	68.3	98.4	54.2
29.61	97.3	70.2	97.3	56.9
26.11	96.1	74.6	95.7	59.3
23.64	94.3	74.0	94.1	61.5
21.74	92.9	74.9	92.7	62.1
20.20	91.6	74.0	91.4	62.1
18.92	90.4	74.2	89.4	63.9
17.81	89.4	73.4	89.1	65.8
16.85	88.8	72.6	88.0	65.8
16.00	87.5	74.5	87.0	66.7
15.23	86.5	74.4	85.5	67.3
14.55	85.5	74.0	84.0	68.2
13.92	84.5	75.3	83.8	70.4
13.35	84.2	75.3	83.2	69.9
12.82	83.6	75.6	82.4	70.9
12.34	82.6	76.7	81.2	72.1
11.88	81.6	76.2	80.4	72.8
11.46	80.9	75.6	79.3	72.0
11.07	79.8	75.2	78.4	71.8
10.70	78.7	74.7	77.3	72.4
10.02	76.9	75.0	75.6	73.0
9.42	75.6	74.2	74.0	72.3
8.88	73.9	72.8	71.3	72.3
8.39	72.2	72.1	69.1	71.9
7.95	70.4	70.4	66.9	70.3
7.54	69.1	68.3	65.2	68.8
7.16	66.5	67.1	63.0	67.7
6.82	64.4	66.7	60.7	66.7
6.20	59.2	63.2	56.1	63.9
5.67	55.4	58.6	52.1	60.6
4.99	49.0	52.5	46.2	55.6

Tabulka 1: Úspěšnost detekce skokové změny trendu pro různé hodnoty SNR

SNR [dB]	ESE [%]	ELBND [%]	LE [%]	Err [%]
39.70	100.0	94.6	99.5	95.2
33.68	100.0	94.6	98.7	95.3
30.16	99.8	94.7	98.1	95.4
27.67	99.5	94.7	96.7	95.3
25.73	99.2	94.6	96.4	95.4
25.07	99.1	94.6	96.3	95.4
24.15	98.9	94.5	96.2	95.2
23.33	98.6	94.3	95.9	95.0
22.82	98.5	94.2	95.7	94.9
22.11	98.1	94.2	95.7	95.1
21.67	97.8	94.3	95.2	94.8
20.66	97.4	94.0	94.8	94.5
19.75	96.9	93.9	93.9	94.2
18.93	96.7	93.8	93.6	94.2
18.19	96.4	93.5	93.1	94.0
17.77	95.7	93.1	92.5	93.6
17.51	95.3	92.7	92.2	93.2
17.12	94.8	92.1	91.6	92.7
16.88	94.5	92.1	91.1	92.4
16.52	94.0	91.8	90.7	92.1
16.29	94.1	91.6	90.3	92.1
15.96	93.5	91.5	90.2	91.7
15.75	92.8	91.4	89.5	91.2
15.24	91.7	91.0	88.4	91.0
15.04	91.4	90.8	88.0	90.8
14.76	90.7	90.0	87.5	90.6
14.48	90.3	89.5	86.9	90.3
14.31	90.2	89.2	86.3	90.1
14.13	90.1	88.9	85.6	89.9
13.88	89.1	88.4	85.5	89.5
10.60	78.8	80.2	75.2	82.3
10.09	76.6	77.9	73.6	80.4
9.62	74.6	75.9	71.8	78.0
9.19	72.1	73.6	69.6	75.6
8.78	69.9	71.6	68.0	73.6
8.41	68.2	70.4	65.7	71.2
8.05	66.5	68.8	64.2	69.1
7.72	63.9	67.4	62.0	67.2
7.40	61.5	64.2	59.6	64.8
3.64	30.1	32.4	30.1	31.4
1.33	11.7	11.2	11.5	9.7

Tabulka 2: Úspěšnost detekce skokové změny parametrů generátoru signálu pro vstupy z rovnoměrného rozdělení $U(-1, 1)$

SNR [dB]	ESE [%]	ELBND [%]	LE [%]	Err [%]
43.86	100.0	96.1	99.9	94.8
34.32	100.0	96.1	98.4	94.9
31.83	99.8	95.8	98.1	94.7
29.89	99.6	96.0	97.7	94.6
28.31	99.4	96.0	97.1	94.6
25.82	99.1	95.7	96.9	94.5
24.80	98.9	95.7	96.4	93.9
23.89	98.5	95.6	96.1	93.6
22.31	97.9	95.0	95.5	92.7
21.96	97.6	94.8	95.3	92.5
21.62	97.5	94.7	95.3	92.4
20.98	96.5	94.2	94.5	91.6
20.39	96.1	94.1	93.9	91.5
19.84	95.8	93.5	93.8	91.2
18.83	95.2	92.7	92.6	90.0
18.37	94.9	92.5	92.3	89.7
17.93	94.7	92.2	91.8	88.5
17.12	93.4	91.0	90.4	87.1
16.39	92.1	89.9	89.4	86.1
15.71	90.1	88.6	87.8	84.6
15.09	88.6	87.1	86.8	83.6
14.52	87.3	86.4	85.8	82.0
13.98	85.0	84.9	84.2	79.5
13.02	82.1	81.7	81.6	76.4
12.16	79.7	78.9	78.8	73.6
11.77	78.6	77.8	77.9	72.3
11.05	75.9	74.8	74.1	69.3
10.88	75.0	74.1	73.6	68.8
10.09	71.3	70.4	69.7	65.5
9.67	68.8	68.7	67.0	63.9
9.27	67.1	65.9	65.0	62.2
9.01	65.8	64.0	63.5	60.9
8.77	64.2	62.0	62.0	59.0
8.10	59.9	58.1	57.8	54.9
7.89	58.2	56.8	56.4	54.1
7.41	55.5	55.1	53.8	50.5
6.97	52.1	52.2	50.8	46.6
6.20	45.9	47.3	45.6	42.1
5.54	41.0	40.7	40.1	37.8
4.98	36.3	36.5	36.5	34.7
4.08	30.0	28.8	28.4	27.3
3.71	27.4	26.0	25.9	24.5
3.10	21.6	22.2	21.4	20.5
2.85	19.9	20.3	19.2	19.4
2.25	15.8	15.4	15.3	15.2
2.09	14.7	14.3	14.0	14.0

Tabulka 3: Úspěšnost detekce skokové změny parametrů generátoru signálu pro vstupy z normálního rozdělení $N(0, 1)$