

# RYCHLÉ ALGORITMY PRO ADAPTIVNÍ DETEKCI NOVOSTI

## Fast algorithms for adaptive novelty detection

*Matouš Cejnek*

*Ústav přístrojové a řídicí techniky, Fakulta strojní, ČVUT v Praze, matousec@gmail.com*

*Abstrakt: Adaptivní algoritmy pro detekci novosti v datech jsou populární nástroj díky své schopnosti kompenzovat některé aspekty nestacionarity data generujících procesů. V tomto článku jsou porovnány dva adaptivní algoritmy pro detekci novosti (Learning Entropy, Error and Learning Based Novelty Detection). Oba nástroje pro detekci novosti jsou testovány na následujících adaptivních algoritmech: NLMS, NLMF, RLS a GNGD. Výsledky experimentální analýzy přináší nové informace o vlivu učících algoritmů na přesnost detekce za použití jejich parametrů. Během experimentální analýzy byly použita syntetická data zatížená různou úrovní šumu a obsahující concept drift.*

*Klíčová slova: detekce novosti, adaptivní filtry, algoritmy*

*Abstract: Adaptive algorithms are a popular tool for novelty detection because their ability to compensate some aspects of nonstationarity in data generating processes. Two adaptive algorithms (Learning Entropy, Error and Learning Based Novelty Detection) are compared in this study. Both adaptive novelty detection tools are tested with following learning algorithms: NLMS, NLMF, RLS and GNGD. The results of the experimental analysis reveals new insights in learning algorithm influence on final detection performance. A synthetic data with various levels of noise and concept drift were used for the experimental analysis.*

*Keywords: Detekce novosti, adaptivní algoritmy, concept drift*

## 1 Úvod

Detekce novosti (*novelty detection*) je klíčové téma v oblasti strojového učení. Označení detekce novosti se používá pro automatizovaný proces vyhledávání neočekávaných hodnot v měřených datech. Tyto neočekávané hodnoty mohou představovat chyby měření, změny v procesu, který generuje data, nebo jakoukoliv jinou větší či menší anomálii. Už z podstaty věci, je často nezbytné aby nástroje pro detekci novosti byly dostatečně rychlé pro nasazení v reálném čase [1]. V současnosti existuje velká množina algoritmů pro detekci novosti, postavených na různých principech. Více informací o různých přístupech k detekci novosti je možné získat například ze srovnávací studie [2].

Tento článek pojednává o dvou specifických algoritmech *Error and learning based novelty detection* (ELBND) [3] a *Learning Entropy* (LE) [4]. Tyto algoritmy spadají do kategorie algoritmů pro adaptivní detekce novosti. Adaptivní znamená, že algoritmus se přizpůsobuje změnám v datech a za novost označuje pouze to, co do dat nezapadá z pohledu krátkodobého kontextu. Délka tohoto kontextu závisí na rychlosti adaptace. Oba studované algoritmy (ELBND a EL) jsou použitelné v kombinaci s libovolným adaptivním modelem, který používá adaptivní parametry. Výhoda detekce novosti na základě krátkého kontextu je ta, že je možné ignorovat dlouhodobější změny. Tyto dlouhodobější změny se mohou vyskytovat v datech přirozeně, aniž by znamenali novost, kterou je potřeba detekovat.

Jako adaptivní nosný model pro detekci novosti, je zde v tomto článku použit lineární adaptivní filtr. Algoritmy, které byly použity pro adaptaci těchto adaptivních filtrů v této studii jsou *Normalized least-mean squares* (NLMS), *Normalized least-mean fourth* (NLMF), *Recursive least squares* (RLS) a *Generalized normalized gradient descent* (GNGD). Tyto algoritmy mají rozdílné vlastnosti a budou více představeny v sekci 2.

Jako data pro provedení experimentální analýzy je použit synteticky vytvořený rozsáhlý data-set. Tento data-set byl vytvořen tak, aby představoval různé výzvy, s kterými je možné se setkat při detekci novosti. Jedna z těchto výzev je *concept drift* [5]. Concept drift (nadále jen drift) je možné popsat jako silnou ne-stacionaritu

dat, která může mít různý původ a různý průběh. Nejčastější případ je graduální drift. V použitém data-setu byl simulován graduální opakující se drift modelovaný jako harmonická vlna přidaná k signálu nesoucímu informace. Další výzva, kterou obsahuje vytvořený data-set, je aditivní šum o různých úrovních. Šum je nejčastější forma znehodnocení dat, proto je použit i v této studii.

Hlavní přínos této studie je otestování a porovnávání více adaptivních algoritmů. Speciálně vhodnost GNGD a NLMF pro detekci novosti pomocí LE a ELBND zatím nikdy nebyla zkoumána.

## 2 Metody

V této sekci jsou vysvětleny metody použité v této studii, včetně postupu křížové validace, která byla použita pro vyhodnocení výsledků této studie.

### 2.1 Adaptivní filtry

Výstup adaptivního filtru  $\tilde{y}(k)$  je definován

$$\tilde{y}(k) = w_1 \cdot x_1(k) + \dots + w_n \cdot x_n(k) = \mathbf{x}^T(k) \mathbf{w}(k), \quad (1)$$

kde  $(\cdot)^T$  je označení pro maticovou transpozici,  $\mathbf{w}(k) = [w_1(k), \dots, w_n(k)]$  je vektor adaptivních vah a  $\mathbf{x}$  je vstupní vektor. Na začátku jsou adaptivní váhy obvykle nastaveny na náhodné hodnoty s normální distribucí, nulovou střední hodnotou a jednotkovou standardní deviací. Vstupní vektor  $\mathbf{x}$  (pro adaptivní filtr velikosti  $n$ ) vypadá následovně

$$\mathbf{x}(k) = [x_1(k), \dots, x_n(k)]. \quad (2)$$

Chyba adaptivního filtru je následovně vyčíslitelná jako

$$e(k) = y(k) - \tilde{y}(k), \quad (3)$$

Jednotlivé algoritmy adaptace užití v této studii jsou vysvětleny níže.

#### NLMS

Adaptivní filtr NLMS [6] je pravděpodobně nejpopulárnější adaptivní filtr. Vektor adaptivních vah NLMS filtru  $\mathbf{w}$  je adaptován na základě pravidla

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \Delta \mathbf{w}(k) = \mathbf{w}(k) + \eta(k) \mathbf{w}(k) e(k), \quad (4)$$

kde  $\eta(k)$  je rychlost učení normalizovaná podle výkonu vstupu

$$\eta(k) = \frac{\mu}{\varepsilon + \|\mathbf{x}(k)\|^2}, \quad (5)$$

kde  $\varepsilon \in \mathbb{R}$  je malá pozitivní konstanta (regularizační výraz) zavedený proto, aby se zachovala stabilita i při vstupech blízkých nule. NLMS algoritmus s tímto regularizačním výrazem je často nazýván  $\varepsilon$ -NLMS.

#### NLMF

Adaptivní filtr NLMF je používán, protože má obecně vyšší schopnost potlačit šum než NLMS filtr [7]. Na druhou stranu zajištění jeho stability je mnohem komplikovanější než u NLMS algoritmu [8, 9]. NLMF adaptace [6] je velice podobná NLMS adaptaci. Vektor adaptivních vah NLMF filtru  $\mathbf{w}$  je adaptován na základě pravidla

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \Delta \mathbf{w}(k) = \mathbf{w}(k) + \eta(k) \mathbf{w}(k) e(k)^3. \quad (6)$$

kde  $\eta(k)$  má stejný význam jako v případě NLMS.

#### GNGD

Adaptivní filtr GNGD [10] je vytvořen rozšířením NLMS adaptivního filtru. Vektor adaptivních vah GNGD filtru  $\mathbf{w}$  je adaptován na základě pravidla

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \Delta \mathbf{w}(k) = \mathbf{w}(k) + \eta(k) \mathbf{w}(k) e(k)^3, \quad (7)$$

kde  $\eta(k)$  má odlišný význam než v NLMS a NLMF. Adaptivní rychlost učení  $\eta(k)$  počítaná podle (5) využívá proměnlivý regularizační výraz  $\varepsilon$ , získaný následovně

$$\varepsilon(k) = \varepsilon(k-1) - \rho \mu \frac{e(k) - e(k-1) \mathbf{x}^T(k) \mathbf{x}(k-1)}{(\|\mathbf{x}(k-1)\|^2 + \varepsilon(k-1))^2}. \quad (8)$$

## RLS

Pravidlo RLS pro výpočet přírůstku adaptivní vah  $\Delta \mathbf{w}(k)$  je definováno následovně

$$\Delta \mathbf{w}(k) = \mathbf{R}(k) \mathbf{x}(k) e(k), \quad (9)$$

kde  $\mathbf{R}(k)$  je inverzí matice k autokorelační matici a je spočítána následovně

$$\mathbf{R}(k) = \frac{1}{\mu} (\mathbf{R}(k-1) - \frac{\mathbf{R}(k-1) \mathbf{x}(k) \mathbf{x}(k)^T \mathbf{R}(k-1)}{\mu + \mathbf{x}(k)^T \mathbf{R}(k-1) \mathbf{x}(k)}). \quad (10)$$

Počáteční hodnota inverzní matice je zvolena

$$\mathbf{R}(0) = \frac{1}{\delta} \mathbf{I}, \quad (11)$$

kde  $\mathbf{I}$  je matice identity a  $\delta$  je malá pozitivní konstanta.

## 2.2 Klasifikace a křížová validace

V této studii je použit podobný klasifikační framework jako v předchozích studiích [11] a [3]. Tento klasifikační framework má za úkol otestovat míru úspěšnosti testovaných metod na velkém množství dat na základně anotovaných segmentů dat. Veškerá data jsou rozdělena na segmenty a každý segment je anotovaný informací, zda-li obsahuje změnu (novou událost) nebo ne. Klasifikační framework následně testuje, zda skutečná anotace dat odpovídá detekovanému stavu pro různé prahy citlivosti detekce. Prah se mění v celém rozsahu spektra - od hodnoty kde jsou všechny segmenty klasifikovány jako nové, až po hodnotu kde jsou všechny segmenty dat klasifikovány jako normální. Z takto získaných dat (množství úspěšných detekcí a množství neúspěšných detekcí) je možné spočítat *specificitu* a *senzitivitu*. Z těchto metrik je následně možné určit *receiver operating characteristic* (ROC) křivku, která je užita pro vizuální porovnání výsledků pro různé nastavení prahu. Výsledky jsou dále také redukovány na jedno číselné metriky popisující přesnost algoritmu - plocha pod ROC křivkou (AUROC) a maximální přesnost. Tyto metriky jsou zvoleny z důvodu, že se dobře doplňují. Zatímco maximální přesnost popisuje ideální podmínky klasifikace (optimální nastavení prahu), tak AUROC popisuje schopnost klasifikace bez ohledu na to jak byl klasifikátor nastaven. Pro splnění základní podmínky pro korektní stanovení ROC křivky bylo zapotřebí mít vyvážený data-set ke klasifikaci. Vyvážení tohoto data-setu bylo dosaženo tím, že všechny segmenty dat byly stejně dlouhé a byl stejný počet pozitivních i negativních segmentů.

## 2.3 Metody detekce novosti

Dvě adaptivní metody detekce novosti jsou studovány v tomto článku - ELBND a LE. Obě metody využívají přírůstky adaptivních vah učícího systému. Metoda LE využívá čistě tyto přírůstky, zatímco ELBND metoda navíc používá i samotnou chybu predikce adaptivního modelu. Přírůstky adaptivních vah jsou s chybou korelovány, takže chyby ovlivňuje výstup jak ELBND, tak LE. Primární rozdíl v těchto metodách leží v tom, že ELBND klade na chybu větší důraz. Detailní popis těchto dvou metod následuje v navazujících podsekcích.

### 2.3.1 Error and learning based novelty detection (ELBND)

Metoda ELBND anotuje každý nový vzorek množstvím obsažené novosti na základě přírůstku adaptivních vah a chyby podle následujícího pravidla

$$\text{ELBND}(k) = \Delta \mathbf{w}(k) e(k). \quad (12)$$

Výstup 12 je vektor, který popisuje míru novosti pro každou adaptivní váhu zvlášť v daném diskretním čase  $k$ . Tento vektor lze dále zobecnit na skalár, pro jednodušší vyhodnocení novosti v daném vzorku. Běžný způsob jak získat z vektoru  $\text{ELBND}(k)$  skalár  $\text{elbnd}(k)$  je funkce maxima z absolutních hodnot

$$\text{elbnd}(k) = \max |\text{ELBND}(k)|. \quad (13)$$

Všimněte si, že tento způsob výpočtu novosti nepotřebuje žádné další parametry, takže problém s optimalizací metody je minimalizován. Na druhou stranu je nutné podotknout, že výstup metody je silně závislý na adaptivním algoritmu (v tomto případě adaptivním filtru). Různé nastavení adaptivního filtru můžou způsobit různé výsledky této metody detekce novosti. Jak bylo ale ukázáno v [3], tak vysoká přesnost predikce není nutně podmínkou pro vysokou přesnost detekce novosti.

### 2.3.2 Learning entropy (LE)

Množství novosti v každém novém vzorku může být spočítáno pomocí Approximate Individual Sample Learning Entropy (AISLE v [4]). Tento způsob je zkráceně nazýván jako Learning Entropy (LE). Výpočet LE je podle následujícího pravidla:

$$LE(k) = \frac{1}{n \cdot n_\alpha} \sum f(\Delta w_i(k), \alpha); \forall \alpha \in \alpha, \quad (14)$$

kde  $n$  je počet adaptivních vah a  $n_\alpha$  je počet citlivostí detekce, které si volí uživatel

$$\alpha = [\alpha_1, \alpha_2, \dots, \alpha_{n_\alpha}]; \alpha_1 < \alpha_2 < \dots < \alpha_{n_\alpha}. \quad (15)$$

Funkce  $f(\Delta w_i(k), \alpha)$  je definována jako

$$f(\Delta w_i(k), \alpha) = \begin{cases} 1, & \text{if } |\Delta w_i(k)| > \alpha \cdot \overline{|\Delta w_{Mi}(k)|} \\ 0, & \text{v ostatních případech} \end{cases} \quad (16)$$

kde  $\overline{|\Delta w_{Mi}(k)|}$  je střední hodnota okna použitého pro výpočet LE. Tato velikost okna by měla být zvolena s ohledem na možnou periodicitu v datech [4]. Počet citlivostí pro detekci  $\alpha$  je volitelný. Doporučení pro volbu hodnot  $\alpha$  podle [4] je volit takové hodnoty, aby výsledná  $LE(k)$  byla nižší než 1 alespoň pro jeden vzorek v datech a zároveň aby maximálně jeden vzorek měl hodnotu  $LE(k) = 0$  na daných datech.

## 3 Experimentální analýza

### 3.1 Použitá data

Pro experimentální analýzu byla použita jedna sada umělých dat. Umělá data byla použita z několika důvodů:

- Řízená generace dat umožňuje přesně anotovat kde jsou změny systému.
- U reálných dat není možné zajistit správnou velikost driftu.
- U reálných dat není možné zajistit přesné množství šumu.
- Reálná data je problematické sehnat ve velkém rozsahu tak, aby splňovala výše uvedené důvody v dostatečné míře.

Tyto umělé data byly vytvořeny skládáním tří různých časových řad:

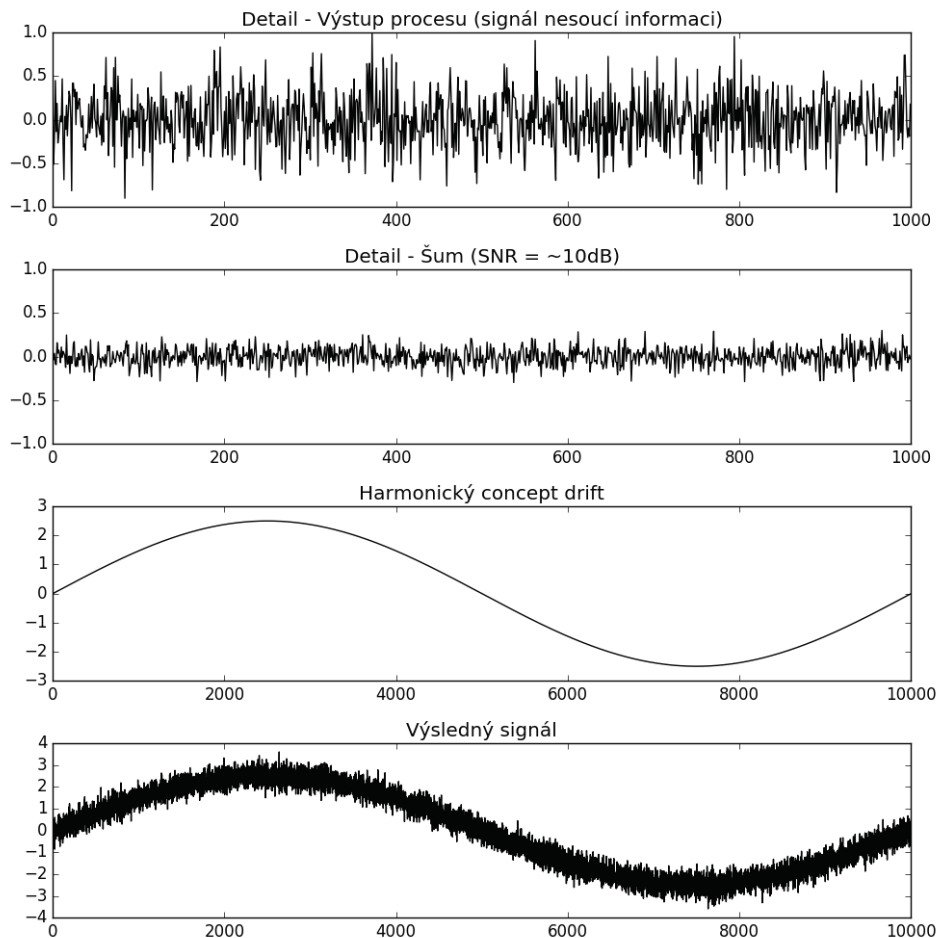
1. Výstup systému obsahující změny, které mají být detekovány.
2. Šum vytvořený tak, aby splňoval požadované parametry
3. *Concept drift* (proměnlivý offset), který byl vytvořený jako harmonická vlna s relativně pomalou periodou.

Signál  $y(k)$ , který nese informaci o změnách, byl vygenerován podle rovnice

$$y(k) = h_1(k)x_1(k) + \dots + h_n(k)x_n(k), \quad (17)$$

kde  $h_i(k)$  jsou parametry procesu a  $x_i(k)$  jsou vstupní proměnné. Jako vstupní proměnné byli použito deset nezávislých časových řad bílého Gaussovského šumu s jednotkovou standardní deviací a nulovou střední hodnotou. Parametry procesu  $h_i(k)$  byly náhodně změněny za jiné parametry každých 500 vzorků. Tyto změny byly ostré v jednom bodě. Nové parametry byly brány z normálního rozložení s standardní deviací 0.5 a nulovou střední hodnotou. celá data obsahují 500 takových změn ( $250 \times 10^3$  vzorků). K vygenerovaným datům byl přičten bílý Gaussovský šum (*Additive white Gaussian noise* (AWGN)). Nakonec byl k datům přičten ještě *concept drift*. Ten byl získán vygenerováním harmonické vlny s periodou ( $10 \times 10^3$  vzorků) vzorků a různou amplitudou. První dvě vlny ( $20 \times 10^3$  vzorků) byly použity k natrénování adaptivních filtrů to ustáleného stavu, zbytek dat byl pak následně použit ke srovnání metod.

Pro lepší představu o skládání dat je přiložen obrázek 1, kde je ukázáno složení ukázkového signálu výše popsaným způsobem (perioda driftu=10000, amplituda driftu=5, SNR přibližně 10dB).



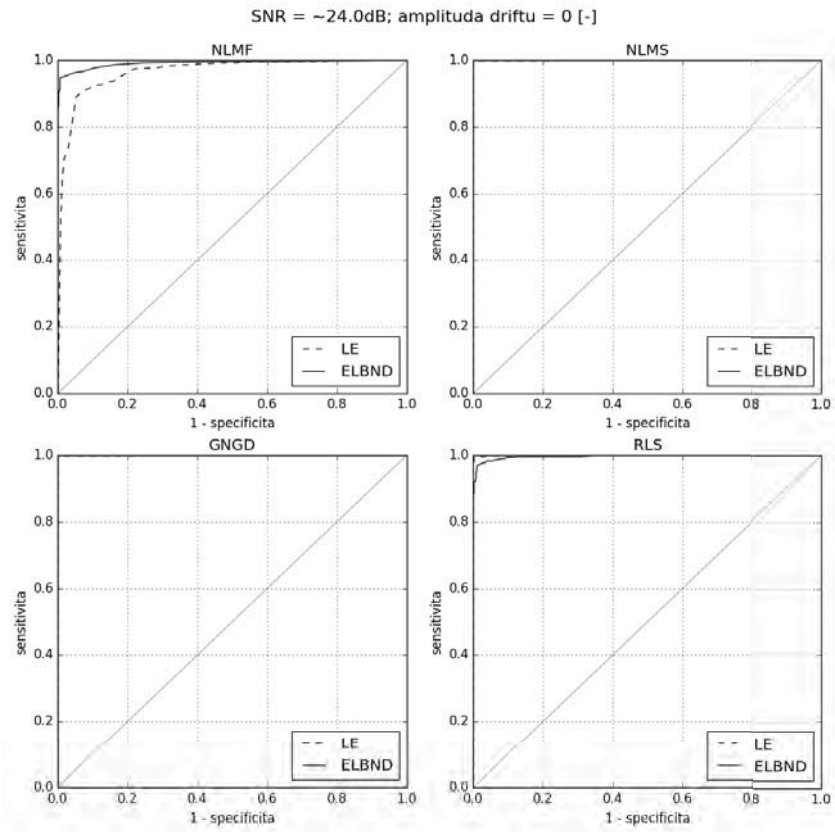
Obr. 1: Časové řady z kterých je složen referenční signál pro adaptivní algoritmy (měřený výstup simulovaného systému).

### 3.2 Výsledná přesnost detekce

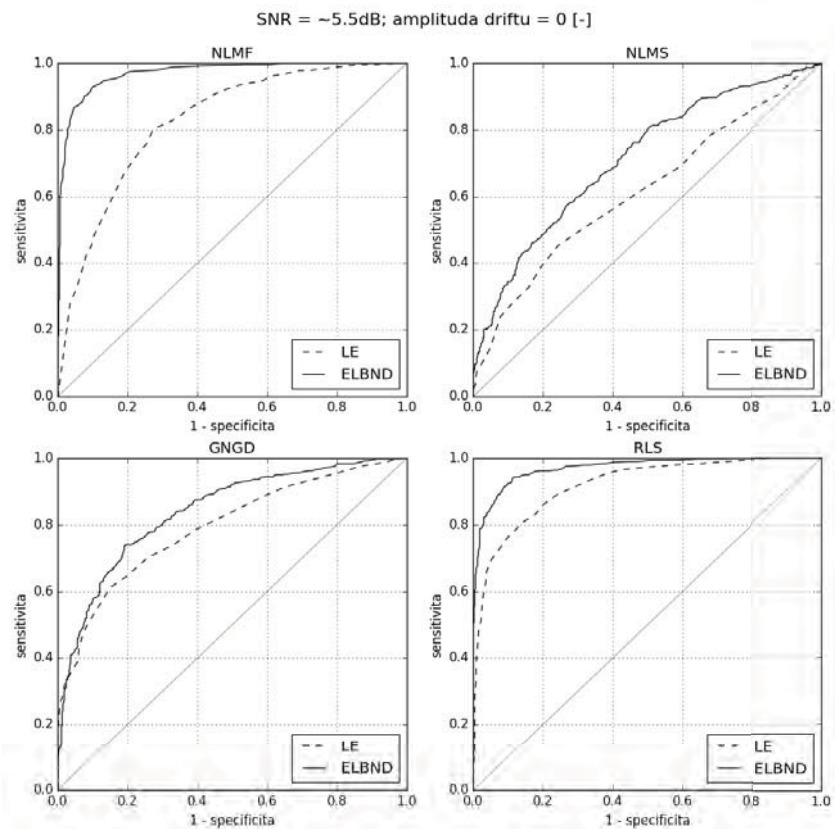
Jak již bylo zmíněno v sekci 2, jako metriky pro číselné srovnání získaných výsledků byli použity AUROC a maximální přesnost (MAX ACC). Samotné výsledné hodnoty pro všechny testované nastavení je možné najít v Tab.1. Kompletní ROC křivky pro vizuální srovnání, z kterých bylo stanoveno AUROC jsou zobrazeny na obrázcích: Obr. 2, Obr. 3, Obr. 4, Obr. 5, Obr. 6 a Obr. 7. Výsledky patrné z obrázků a tabulky je možné shrnout do následujících bodů:

- Poznatek 1. Při vyšší úrovni šumu obecně dosahuje lepších výsledků ELBND než LE.
- Poznatek 2. Při vyšší úrovni driftu dosahuje lepších výsledků LE než ELBND.
- Poznatek 3. Adaptivní algoritmy NLMS a GNGD se zdají být vhodnější pro použití s detekcí pomocí LE, zatímco algoritmy RLS a NLMF se zdají být vhodnější pro ELBND.

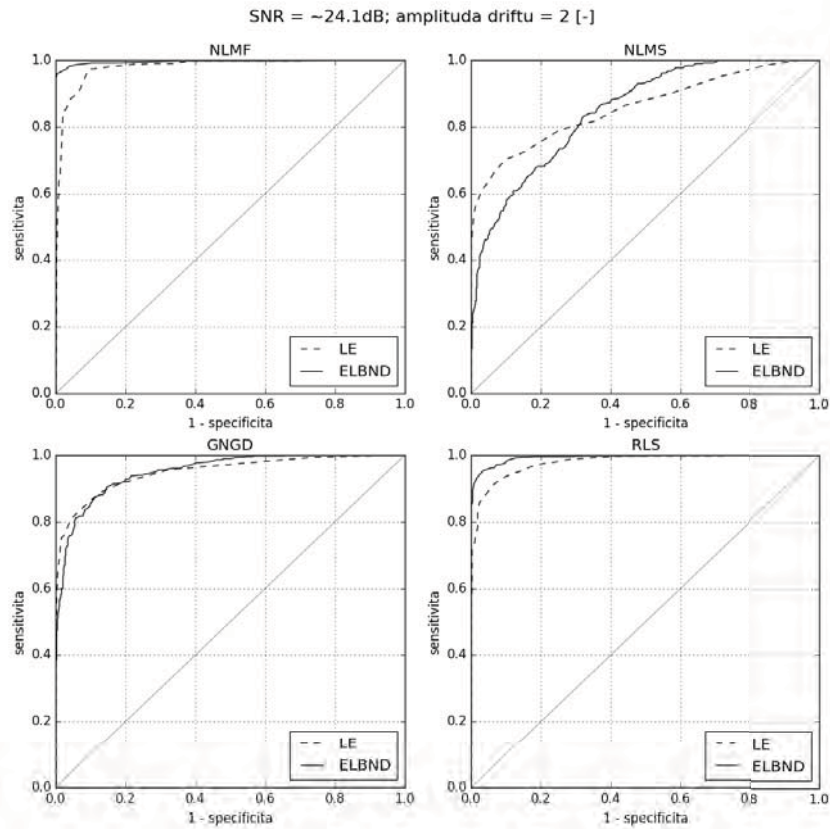
Získané výsledky (Poznatek 1., Poznatek 2.) potvrzují výsledky naší předchozí práce [11] a [3]. Poznatek 3. je nový, jelikož algoritmy GNDG a NLMF předtím nebyly nikdy testovány v tomto kontextu.



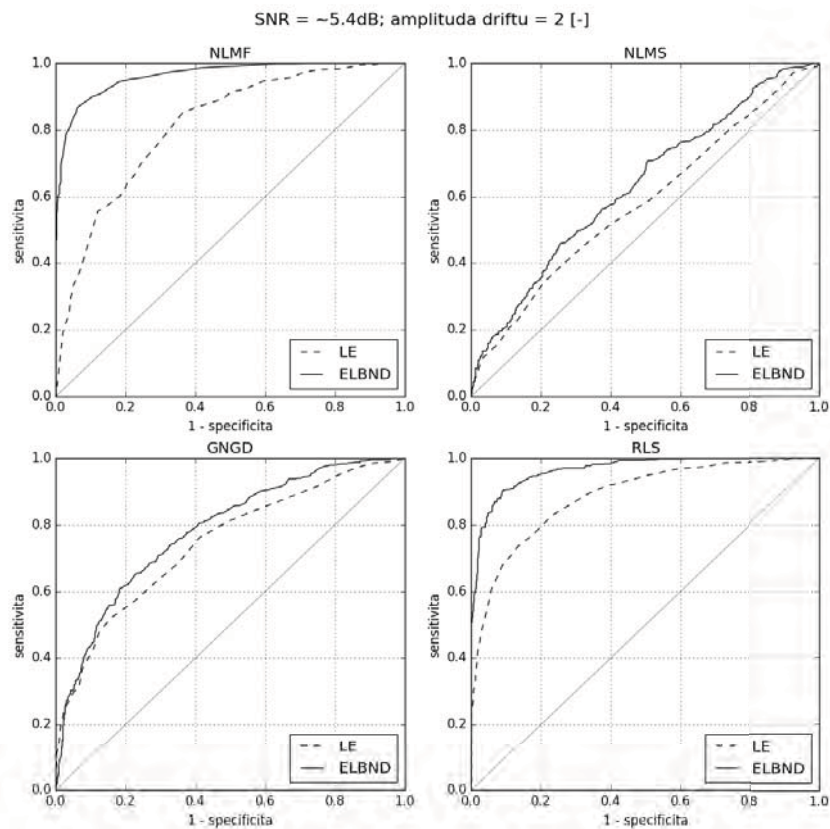
Obr. 2: ROC křivka pro výsledky na datech bez driftu a s nízkou úrovní šumu. Prázdné grafy představují výsledky s nulovou nebo téměř nulovou chybou.



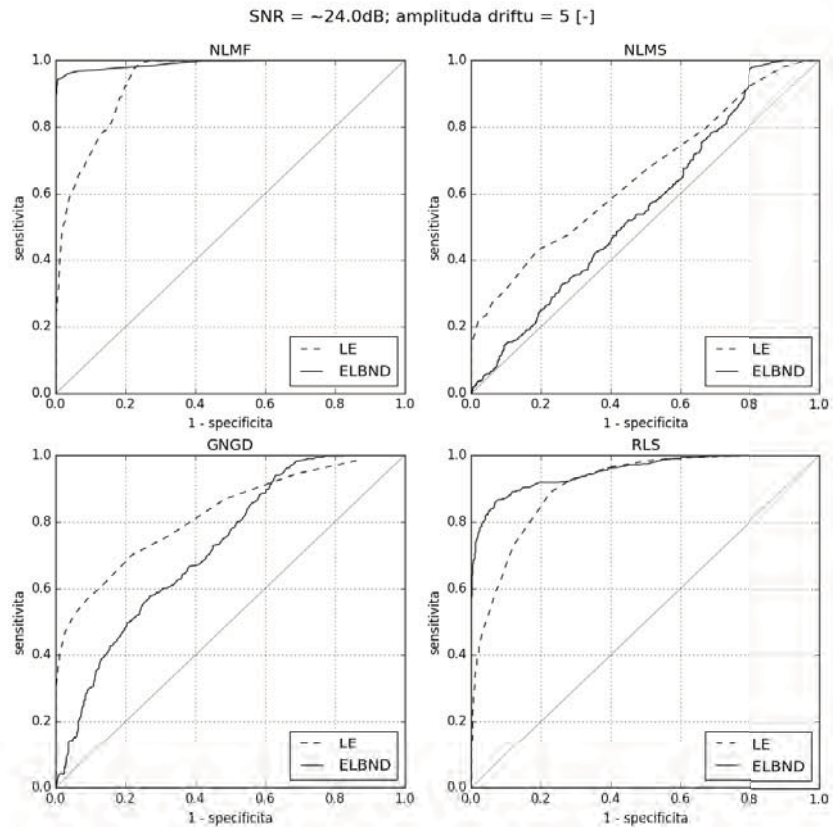
Obr. 3: ROC křivka pro výsledky na datech bez driftu a s vysokou úrovní šumu. Při této konfiguraci dosahuje ve všech případech lepších výsledků ELBND než LE.



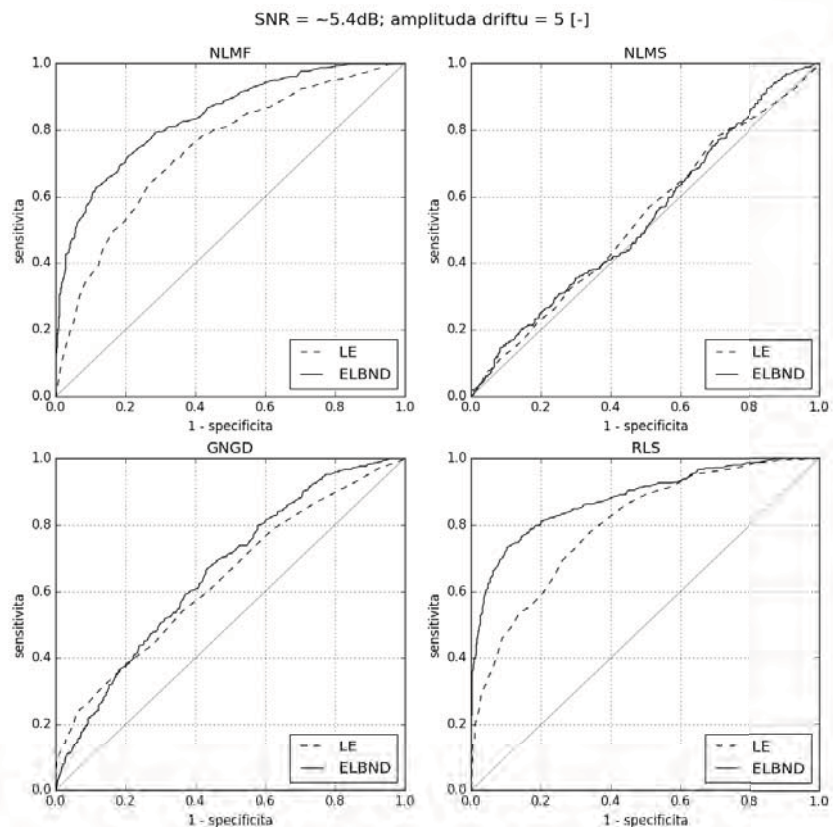
Obr. 4: ROC křivka pro výsledky na datech se středně velkým driftem a s nízkou úrovní šumu. Při této konfiguraci nelze jednoznačně říci zda je lepší LE nebo ELBND. Výsledky jsou závislé na použitém adaptivním algoritmu.



Obr. 5: ROC křivka pro výsledky na datech se středně velkým driftem a s vysokou úrovní šumu. Při této konfiguraci dosahuje lepších výsledků ELBND.



Obr. 6: ROC křivka pro výsledky na datech s velkým driftem a s nízkou úrovní šumu. Při této konfiguraci nelze jednoznačně říci zda je lepší LE nebo ELBND. Výsledky jsou závislé na použitém adaptivním algoritmu.



Obr. 7: ROC křivka pro výsledky na datech s velkým driftem a s vysokou úrovní šumu. Při této konfiguraci jsou výsledky podobné, s tím že pro některé adaptivní algoritmy funguje lépe ELBND.



Tab. 1: Tabulka výsledných všech výsledných hodnot přenosti detekce s EL a ELBND.

Adaptace	Detekce	AUROC [%]	MAX ACC [%]
SNR = 5.4dB; amplituda driftu = 5 [-]			
NLMF	LE	73.9	68.6
	ELBND	83.7	75.7
NLMS	LE	53.1	53.9
	ELBND	53.6	53.5
GNGD	LE	63.6	58.9
	ELBND	65.8	61.7
RLS	LE	79.6	71.9
	ELBND	87.6	81.5
SNR = 24.0dB; amplituda driftu = 5 [-]			
NLMF	LE	93.4	87.6
	ELBND	99.0	96.9
NLMS	LE	65.5	62.0
	ELBND	56.2	58.6
GNGD	LE	81.9	74.1
	ELBND	72.0	65.5
RLS	LE	90.3	83.1
	ELBND	95.0	89.5
SNR = 5.4dB; amplituda driftu = 2 [-]			
NLMF	LE	81.0	74.4
	ELBND	96.1	90.3
NLMS	LE	57.9	56.7
	ELBND	63.1	60.2
GNGD	LE	74.2	68.1
	ELBND	77.9	71.3
RLS	LE	88.3	80.1
	ELBND	96.4	90.6
SNR = 24.1dB; amplituda driftu = 2 [-]			
NLMF	LE	97.4	94.0
	ELBND	99.5	97.9
NLMS	LE	85.8	80.2
	ELBND	84.7	75.5
GNGD	LE	94.8	88.3
	ELBND	94.9	88.0
RLS	LE	97.7	92.4
	ELBND	99.3	96.0
SNR = 5.5dB; amplituda driftu = 0 [-]			
NLMF	LE	82.3	76.2
	ELBND	96.7	91.5
NLMS	LE	61.5	60.5
	ELBND	71.3	65.3
GNGD	LE	79.1	73.1
	ELBND	83.8	77.3
RLS	LE	91.5	83.3
	ELBND	96.8	91.4
SNR = 24.0dB; amplituda driftu = 0 [-]			
NLMF	LE	96.1	91.9
	ELBND	99.0	97.1
NLMS	LE	99.9	99.9
	ELBND	100.0	100.0
GNGD	LE	100.0	99.9
	ELBND	100.0	100.0
RLS	LE	100.0	99.7
	ELBND	99.6	97.9

### 3.3 Časová náročnost

Oba algoritmy EL i ELBND mohou využívat parametry učících algoritmů ve stejné podobě. Tak to bylo realizované i v této práci. Z tohoto důvodu nemá cenu zahrnovat výpočetní náročnost užitých adaptivních algoritmů do této studie. Samotná výpočetní náročnost ELBND a LE je závislá na implementaci nižších funkcí, které využívají (průměrná hodnota pole, dělení desetinných čísel, maximální hodnota pole). Z tohoto důvodu byla dána přednost měření rychlosti namísto určování počtu operací na iteraci. Každopádně už z pravidel které používají oba algoritmy pro svojí detekci je evidentně nižší výpočetní náročnost u algoritmu ELBND než u LE.

Měřené časy byly stanoveny na osobním počítači. Čas byl měřen kumulativně pro všechny simulace, s tím že pro každou simulaci byla změřena ELBND a LE před tím než se přešlo na další simulaci. Toto nastavení bylo provedeno proto, aby se rovnoměrně distribuovalo kolísání výkonu počítače mezi oba algoritmy (ve srovnáním s nastavením kdy by se provedly v jeden čas všechny simulace pro ELBND a následně v další čas všechny simulace pro LE). Naměřený kumulativní čas pro všechny simulace v této studii je 2.963s a pro ELBND a 184.631s pro LE. Pro dané podmínky ELBND dosahovala přibližně 62x vyšší rychlost. Je potřeba si ale uvědomit že tato rychlost závisí silně na zvoleném nastavení. Obecně lze předpokládat že užití kratšího okna pro porovnání vah u LE by výrazně snížilo rychlostní rozdíl. Na druhou stranu při užití delšího vstupního vektoru by se rozdíl v rychlosti navýšil.

## 4 Závěr a diskuze

V této práci byly porovnány algoritmy LE a ELBND pro detekci novosti na datech zatížených šumem a opakujícím se harmonickým driftem. Práce navázala na naší předchozí práci a potvrdila výsledky již dříve získané. V neposlední řadě tato práce poukázala na vliv volby adaptivního algoritmu na samotné výsledky detekce novosti. Podle získaných výsledků se zdá, že algoritmy NLMF a RLS fungují lépe s ELBND zatímco GNGD a NLMS pracují obecně lépe s LE. Tento rozdíl by se dal vysvětlit tím že NLMS a GNGD jsou gradientní metody, které jsou navzájem velice podobné. Algoritmus NLMF je také gradientní metoda, ale je od výše zmíněných výrazně odlišný tím, že ve svém učení dává větší důraz na chybu modelu ( $e(k)^4$  namísto  $e(k)^2$ ). Algoritmus RLS není gradientní a je zásadně jiný od všech ostatních. Tento poznatek je zajímavým námětem na další výzkum.

## Poděkování

Tento projekt byl podpořen grantem *SGS18/177/OHK2/3T/12*. Všechny simulace byly provedeny v jazyce *Python*. Zdrojové kódy je možné dostat na požádání od autora.

## Literatura

- [1] Subutai Ahmad, Alexander Lavin, Scott Purdy, and Zuha Agha. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing*, 262:134 – 147, 2017.
- [2] Marco AF Pimentel, David A Clifton, Lei Clifton, and Lionel Tarassenko. A review of novelty detection. *Signal Processing*, 99:215–249, 2014.
- [3] Matouš Cejnek and Ivo Bukovsky. Concept drift robust adaptive novelty detection for data streams. *Neurocomputing*, 2018.
- [4] Ivo Bukovsky. Learning entropy: Multiscale measure for incremental learning. *Entropy*, 15(10):4159–4187, 2013.
- [5] Jeffrey C Schlimmer and Richard H Granger. Beyond incremental processing: Tracking concept drift. In *AAAI*, pages 502–507, 1986.
- [6] Ali H Sayed. *Fundamentals of adaptive filtering*. John Wiley & Sons, 2003.
- [7] V Sreenivasa Arun Kumar et al. Comparison of stable nlmf and nlms algorithms for adaptive noise cancellation in ecg signal with gaussian, binary and uniform signals as inputs. *International Journal of Engineering Research and Applications*, 4(8):28–33, 2014.
- [8] Victor H Nascimento and José Carlos M Bermudez. Probability of divergence for the least-mean fourth algorithm. *IEEE transactions on signal processing*, 54(4):1376–1385, 2006.
- [9] Pedro Inacio Hubscher and José Carlos M Bermudez. An improved statistical analysis of the least mean fourth (lmf) adaptive algorithm. *IEEE transactions on Signal Processing*, 51(3):664–671, 2003.

- [10] Danilo P Mandic. A generalized normalized gradient descent algorithm. *IEEE signal processing letters*, 11(2):115–118, 2004.
- [11] Matous Cejnek and Ivo Bukovsky. Influence of type and level of noise on the performance of an adaptive novelty detector. In *2017 IEEE 16th International Conference on Cognitive Informatics & Cognitive Computing (ICCI\* CC)*, pages 373–377. IEEE, 2017.



**Selected article from**  
**Tento dokument byl publikován ve sborníku**

**Nové metody a postupy v oblasti přístrojové techniky,  
automatického řízení a informatiky 2018**  
**New Methods and Practices in the Instrumentation,  
Automatic Control and Informatics 2018**  
**28. 5. – 30. 5. 2018, Příbram - Podlesí**

**ISBN 978-80-01-06477-1**

**Web page of the original document:**  
<http://control.fs.cvut.cz/nmp>  
<http://iat.fs.cvut.cz/nmp/2018.pdf>

**Obsah čísla/individual articles:**  
<http://iat.fs.cvut.cz/nmp/2018/>