

ANALÝZA VLASTNOSTÍ ALGORITMU IMMUNOS-99

Luděk Kopáček

Univerzita Pardubice, Fakulta ekonomicko-správní, Ústav systémového inženýrství a informatiky

Abstract: *The paper presents Immunos-99 as a novel approach to classification task. At first the analysis of behaviour for key steps of training phase is presented. Based on the results of this analysis is designed new version of Immunos-99 with parameterized dynamical thresholding. Algorithm Immunos-99 is used for municipal creditworthiness modelling and sensitivity analysis of parameters setting for this task is provided. Finally the comparison of results is presented.*

Keywords: *Artificial immune system, Immunos-99, sensitivity analysis, municipal creditworthiness modelling.*

1. Úvod

Výpočtová inteligence obsahuje několik oblastí, které byly inspirovány biologickými systémy. Hlavní motivací bylo využití přirozených vlastností biologických systémů pro řešení úloh pomocí výpočtové inteligence. Umělé imunitní systémy (Artificial Immune System – AIS) [8] jsou inspirovány biologickými pochody, které zajišťují obranyschopnost živočichů proti virům a bakteriím. Tyto pochody jsou souhrnně nazývány imunitní systém organismu. Jejich základním úkolem je rozlišovat mezi vlastními buňkami a cizorodými buňkami, viry, apod. Látky, které je imunitní systém schopen rozeznat, se nazývají antigeny. Pro umělé imunitní systémy se využívá principů tzv. specifické imunity, která se vyvíjí na základě zkušeností organismu s cizorodými látkami (tzv. naučená imunita). Základními prvky specifické imunity jsou T-lymfocyty, B-lymfocyty a protilátky. Pro AIS se často nerozlišují a používají se pouze protilátky.

Mezi algoritmy založené na principech AIS patří i skupina algoritmů Immunos, která je určena pro úlohy klasifikace. Prvním představitelem této skupiny populačně založených algoritmů je algoritmus Immunos-81 [7], na základě kterého byly v [5] navrženy varianty Immunos-1, Immunos-2 a Immunos-99. Dle [7] bylo cílem Immunos-81 vytvořit algoritmus s jednoduchou interní reprezentací, schopností zobecnit vstupní data s nominálními, spojitými proměnnými a schopností pracovat s velkým množstvím vzorů. Mezi vlastnosti tohoto algoritmu lze také zahrnout predikovatelnou dobu učení a možnost průběžného učení.

Algoritmus Immunos-81 byl navržen pro řešení několika úloh najednou a pracuje ve dvou krocích (jak fáze učení, tak fáze klasifikace). První krok je zařadit neznámý antigen k příslušné úloze a druhý krok spočívá v naučení nebo klasifikaci neznámého antigenu v příslušné úloze. Popis algoritmu Immunos-81 v [7] byl soustředěn na popis prvního kroku, který ale může být realizován několika instancemi stejného algoritmu [5], kde každá instance řeší konkrétní problém. Vzhledem k nereprodukovatelnosti výsledků vlivem částečně nedostačujícího popisu v [7] byl vytvořen algoritmus Immunos-1 [5], který je založen na principech použitých pro Immunos-81. Dále byly v [5] navrženy varianty Immunos-2 a Immunos-99.

Základním rozdílem mezi algoritmem Immunos-99 a ostatními zástupci této skupiny (Immunos-1 a Immunos-2) je v tom, že algoritmus Immunos-99 vyžaduje nastavení parametrů. Toto nastavení je klíčové pro kvalitu dosažených výsledků v rámci klasifikace.

Tato práce je zaměřena na analýzu chování algoritmu Immunos-99 a na návrh nastavení parametrů pro úlohu modelování bonity obcí.

2. Analýza vlastností algoritmu Immunos-99

Algoritmus Immunos-99 vychází nejen ze svých předchůdců, algoritmu Immunos-81[7] a Immunos-1, ale zahrnuje i některé přístupy z algoritmu CLONALG [3], [4], [5], [9]. Detailní popis algoritmu Immunos-99 lze nalézt v [5] nebo [14]. Vstupní trénovací množina je v rámci fáze učení transformována pomocí principů AIS a výsledkem je tzv. populace paměťových protilátek, která je následně využita v rámci fáze klasifikace. Fáze klasifikace je stejná jako v případě algoritmu Immunos-1. Zásadní vliv má ale fáze učení, která je řízena nastavením tří parametrů: počtem generací pro fázi učení n_{gen} , počáteční velikostí populace protilátek n_{init} a úrovní potlačení protilátek t . V tomto článku budou uvedeny klíčové body fáze učení tohoto algoritmu, detailní popis tohoto algoritmu ve stejné notaci lze nalézt v [14].

Ve fázi učení algoritmu Immunos-99 je každá protilátka \mathbf{ab}_j ohodnocena hodnotou $score(\mathbf{ab}_j, \mathbf{ag}_x)$ dle vztahu [5], [14]

$$score(\mathbf{ab}_j, \mathbf{ag}_x) = |Ab_i| - index_j, \quad (1)$$

kde $index_j$ je pořadové číslo protilátky $\mathbf{ab}_j \in Ab_i$ v seznamu setříděném podle hodnoty $D(\mathbf{ab}_j, \mathbf{ag}_x)$ v rostoucím pořadí (číslováno od nuly) a $|Ab_i|$ je počet protilátek v množině Ab_i . Pro protilátku \mathbf{ab}_j s nejmenší hodnotou metriky $D(\mathbf{ab}_j, \mathbf{ag}_x)$ (nejvíce podobný antigen \mathbf{ag}_x a protilátka \mathbf{ab}_j) je hodnota $score(\mathbf{ab}_j, \mathbf{ag}_x) = |Ab_i|$ (největší), pro protilátku \mathbf{ab}_k s největší hodnotou metriky $D(\mathbf{ab}_j, \mathbf{ag}_x)$ (nejméně podobný antigen \mathbf{ag}_x a protilátka \mathbf{ab}_j) je hodnota $score(\mathbf{ab}_j, \mathbf{ag}_x) = 1$. Hodnota metriky $D(\mathbf{ab}_j, \mathbf{ag}_x)$ je v případě reálných hodnot atributů antigenů a protilátek dána jako

$$D(\mathbf{ab}_j, \mathbf{ag}_x) = \sqrt{\sum_{i=1}^l (ab_{i,j} - ag_{i,x})^2}, \quad (2)$$

kde $ab_{i,j}$ je i -tý atribut j -té protilátky \mathbf{ab}_j a $ag_{i,x}$ je i -tý atribut antigenu \mathbf{ag}_x .

Hodnota $score(\mathbf{ab}_j, \mathbf{ag}_x)$ dle (1) tak vyjadřuje kvalitu protilátky $\mathbf{ab}_j \in Ab_i$ ve vztahu k danému antigenu \mathbf{ag}_x . Na základě hodnot $score(\mathbf{ab}_j, \mathbf{ag}_x)$ pro jednotlivé antigeny je pak stanovena hodnota kvality dané protilátky \mathbf{ab}_j ve vazbě k celé trénovací množině Ag jako

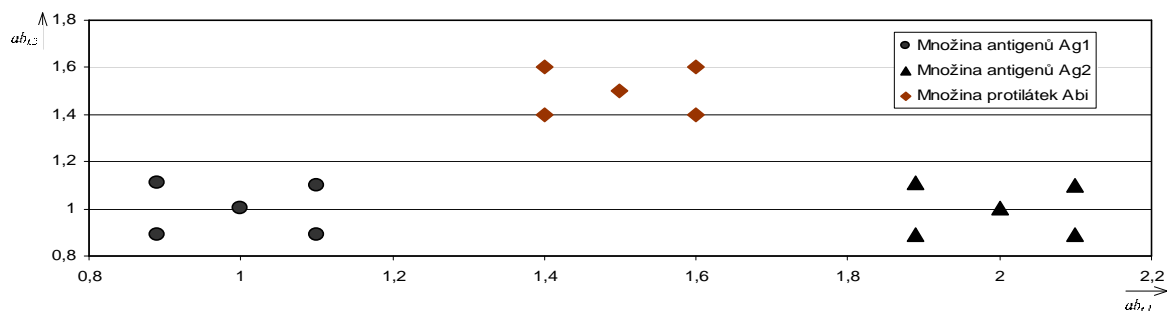
$$fit_j = fitness(\mathbf{ab}_j, Ag) = \frac{correct}{incorrect}, \quad (3)$$

$$correct = \sum_{\substack{\mathbf{ag}_x \in Ag \\ class(\mathbf{ag}_x) = class(\mathbf{ab}_j)}} score(\mathbf{ab}_j, \mathbf{ag}_x), \quad (4)$$

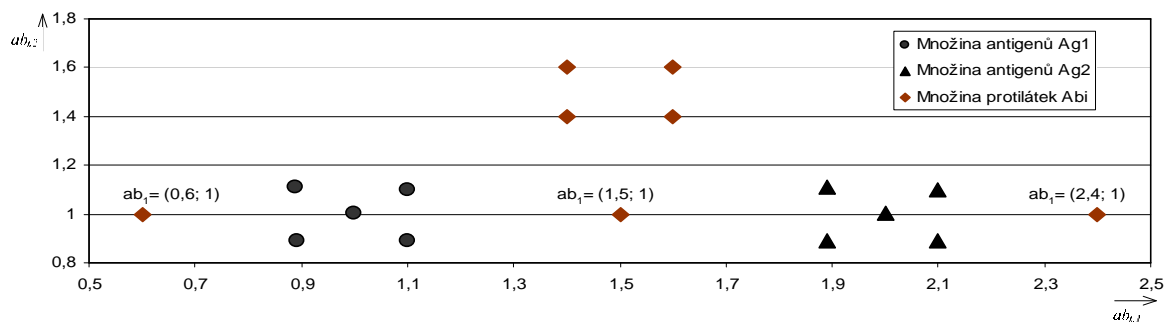
$$incorrect = \sum_{\substack{\mathbf{ag}_x \in Ag \\ class(\mathbf{ag}_x) \neq class(\mathbf{ab}_j)}} score(\mathbf{ab}_j, \mathbf{ag}_x). \quad (5)$$

Hodnota kvality fit_j pro protilátku $\mathbf{ab}_j \in Ab_i$, dle (3) je závislá mimo jiné i na velikosti dané množiny protilátek Ab_i . Tento jev je způsoben závislostí hodnoty hodnot $score(\mathbf{ab}_j, \mathbf{ag}_x)$ nejen na protilátce \mathbf{ab}_j a antigenu \mathbf{ag}_x , ale také na velikosti množiny protilátek Ab_i , ze které protilátka $\mathbf{ab}_j \in Ab_i$ pochází (1). Pro demonstraci tohoto vlivu bude použit příklad z obr. 1, který obsahuje dvě množiny antigenů Ag_1 a Ag_2 , každá po 5 antigenech. Každý antigen je

charakterizován dvěma parametry, tak aby bylo možné příklad znázornit graficky. Dále necht' je dána množina protilátek Ab_i , která obsahuje také 5 protilátek. Třída odpovídající množině Ab_i je stejná jako třída odpovídající množině antigenů Ag_1 . Rozložení prvků množiny protilátek Ab_i a množin antigenů Ag_1 a Ag_2 ukazuje obr. 1. Hodnota proměnné $y = 1$ pro protilátku ab_1 bude konstantní a hodnota proměnné x se bude postupně měnit, jak znázorňuje obr. 2.

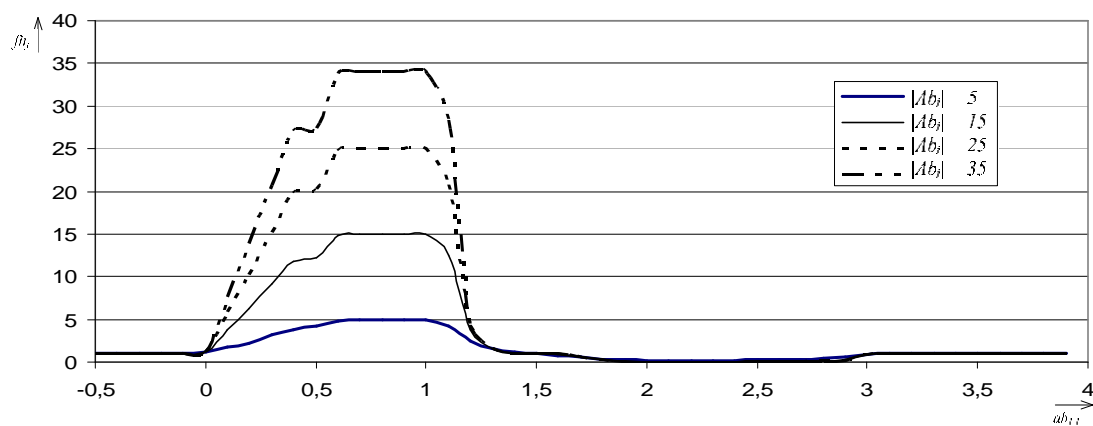


Obr. 1: Rozložení antigenů jednotlivých tříd pro demonstraci výpočtu kvality protilátek

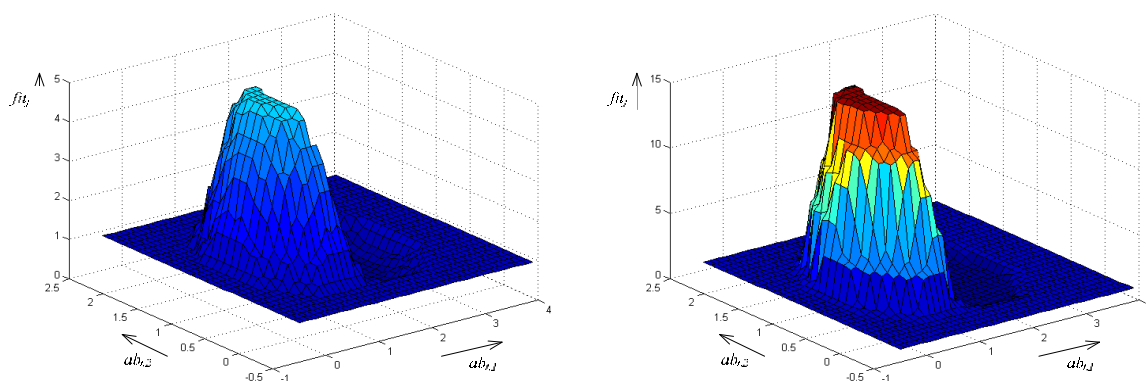


Obr. 2: Znázornění změny parametrů protilátky ab_1

Na obr. 3 je znázorněn průběh hodnoty kvality protilátky ab_1 v závislosti na změně parametru $ab_{1,1}$. Jednotlivé křivky pak ukazují vliv počtu protilátek množiny Ab_i na tvar této závislosti. Jak je z uvedených průběhů patrné, tak za jinak nezměněných podmínek dochází při rostoucím počtu protilátek v množině Ab_i k růstu maximální hodnoty kvality protilátky. Na obr. 4 je znázorněna stejná závislost, ale ve vazbě na změnu hodnoty proměnné $ab_{1,1}$ a $ab_{1,2}$ protilátky ab_1 pro $|Ab_i| = 5$ obr. 4 a) a $|Ab_i| = 15$ obr. 4 b) (viz změna měřítka osy z).



Obr. 3: Vliv počtu protilátek populace Ab_i na hodnotu kvality protilátky $ab_j - 2D$



a) b)

Obr. 4: Vliv počtu protilátek populace Ab_i na hodnotu kvality protilátky ab_j – 3D, a) počet protilátek množiny $|Ab_i| = 5$, b) počet protilátek množiny $|Ab_i| = 15$

Závislost hodnoty kvality fit_j pro protilátku ab_j na rozložení protilátek v dané množině protilátek Ab_i a na velikosti této množiny není jediná. Kvalita fit_j protilátky ab_j dle vztahu (3) je funkcí těchto parametrů:

- protilátky ab_j ,
- rozložení a velikost množiny Ab_i , do které ab_j náleží, tj. $ab_j \in Ab_i$ a
- rozložení a velikosti množiny antigenů Ag .

Nepřímo se výpočet kvality protilátky ab_j váže na třídu c_i , která odpovídá množině Ab_i a tím i dané protilátce. Kromě velikosti množiny Ab_i je pro stanovení možného rozsahu hodnoty kvality jedince také určující i poměr počtu antigenů, s třídou odpovídající protilátce ab_j , k počtu antigenů, s jinou třídou, než odpovídá ab_j

$$r_{ag}(c_i) = \frac{|Ag_i|}{\sum_{\substack{\forall Ag_j \\ j \neq c_i}} |Ag_j|}. \quad (6)$$

Poměr $r_{ag}(c_i)$ obsahuje v čitateli počet antigenů trénovací množiny odpovídající třídě c_i a v jmenovateli je počet všech antigenů, které neodpovídají třídě c_i . Na základě způsobu výpočtu hodnoty kvality protilátky fit_j dle vztahu (3) a výpočtu hodnoty $score(ab_j, ag_x)$ dle vztahu (1) lze odvodit vztahy pro minimální a maximální hodnotu kvality jedince ve vztahu k uvedenému poměru $r_{ag}(c_i)$ a počtu protilátek v množině $|Ab_i|$ odpovídající třídě c_i

$$fit_{\min}(c_i) = \frac{r_{ag}(c_i)}{|Ab_i|}, \quad (7)$$

$$fit_{\max}(c_i) = r_{ag}(c_i) \times |Ab_i|. \quad (8)$$

Z uvedených vztahů lze odvodit tyto závislosti:

- Vliv počtu antigenů stejné třídy $|Ag_i|$ – Čím větší je počet antigenů stejné třídy, tím větší je hodnota poměru $r_{ag}(c_i)$ a tím větší jsou hodnoty $fit_{min}(c_i)$ a $fit_{max}(c_i)$.
- Vliv počtu antigenů jiné třídy – Čím větší je počet antigenů jiné třídy, tím menší je hodnota poměru $r_{ag}(c_i)$ a tím menší jsou hodnoty $fit_{min}(c_i)$ a $fit_{max}(c_i)$.
- Vliv počtu protilátek dané třídy $|Ab_i|$ – Čím větší je počet protilátek stejné třídy, tím menší je hodnota $fit_{min}(c_i)$ a tím větší je hodnota $fit_{max}(c_i)$.

Demonstrace uvedených závislostí je pro daný příklad uvedena v tab. 2. Tento jev se velmi nepříznivě odráží na kvalitě dosažených výsledků v případě, kdy trénovací množina obsahuje nerovnoměrně rozložené prvky mezi jednotlivými třídami. V případě, kdy je rozložení trénovací množiny rovnoměrné, tak se projevuje pouze vliv počtu protilátek odpovídající množiny. Tento vliv již není tak významný, protože rozložení protilátek v rámci jednotlivých množin přibližně kopíruje rozložení v trénovací množině.

Tab. 2: Demonstrace závislosti minimální a maximální hodnoty kvality protilátky

$ Ab_i $	$ Ag_i $	$ Ag_j $, kde $j \neq c_i$	$f_{min}(c_i)$	$f_{max}(c_i)$
5	5	5	0.2	5.0
5	5	10	0.1	2.5
5	5	15	0.067	1.67
5	5	5	0.2	5.0
5	10	5	0.4	10.0
5	15	5	0.6	15.0
5	5	5	0.2	5.0
10	5	5	0.1	10.0
15	5	5	0.067	15.0

Uvedená závislost oboru hodnot pro kvalitu protilátek se nepříznivě projevuje při stanovení parametru t , který je nutné nastavit v závislosti na oboru hodnot pro kvalitu protilátek. Množina Ab_i je tedy po eliminaci [14] s absolutním nastavením prahu t vyjádřena takto

$$Ab'_i = \{ \mathbf{ab}_j | \mathbf{ab}_j \in Ab_i; fit_j > t \}, \quad (9)$$

kde Ab_i je množina protilátek před eliminací a fit_j je hodnota kvalita protilátky dle vztahu (3) vzhledem k množině antigenů Ag .

Pokud mají dvě třídy v trénovací množině rozdílný počet vzorů, jsou obory hodnot pro dané třídy také rozdílné. Stanovení jedné hodnoty prahu pro jednu ze tříd, může znamenat plné potlačení protilátek v jiné třídě. V takovém případě je stanovení jedné hodnoty prahu pro všechny třídy nemožné. Možným řešením by bylo zavedení více prahovacích hodnot, kde každá hodnota prahu by se vázala k jedné třídě. Tato možnost je však z hlediska nastavení nevhodná. Absolutní nastavení hodnoty parametru t nelze vzhledem k uvedeným skutečnostem vůbec doporučit.

3. Návrh parametrizované verze dynamického prahování algoritmu Immunos-99

Jak bylo uvedeno v předchozí kapitole, tak nelze absolutní prahování dle (9) s nastavenou hodnotou prahu t pro algoritmus Immunos-99 doporučit. Tento způsob prahování je zejména nevhodný v situaci, kdy je v trénovací množině různý počet vzorů mezi jednotlivými třídami.

Pro tento případ je v originální variantě algoritmu Immunos-99 [5] možné využít tzv. dynamického prahování. Pokud je hodnota $t = -1$, je jako práh použita průměrná hodnota kvality fit_j pro protilátky \mathbf{ab}_j z dané množiny protilátek Ab_i . Množina protilátek po eliminaci Ab_i [14] je v tomto případě vyjádřena následujícím způsobem

$$Ab'_i = \{ \mathbf{ab}_j \mid \mathbf{ab}_j \in Ab_i; fit_j > \min(fit_{avg}, 1) \}, \quad (10)$$

$$fit_{avg} = \frac{\sum_{\forall \mathbf{ab}_j \in Ab_i} fit_j}{n_{Ab_i}}, \quad (11)$$

kde n_{Ab_i} je počet protilátek v množině Ab_i . Při tomto způsobu potlačení protilátek je pro každou třídu vypočtena průměrná hodnota kvality protilátek dané třídy a potlačení jsou jedinci s hodnotou kvality menší než je vypočtený průměr (11). Při tomto způsobu prahování se výše uvedený vliv rozdílného oboru hodnot kvality protilátek pro jednotlivé třídy eliminuje. Na druhou stranu se tak ztrácí možnost ovlivnit úroveň potlačení vlivem nastaveného prahu.

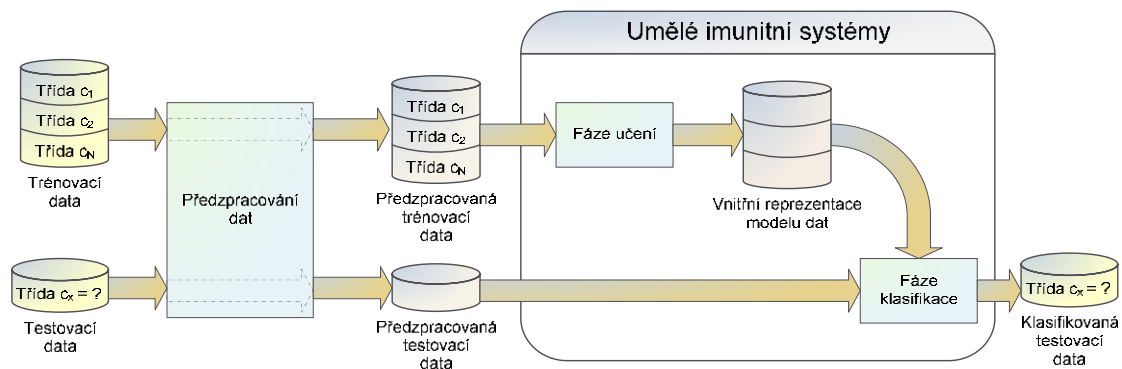
Aby při dynamickém prahování nedošlo ke ztrátě možnosti ovlivnit úroveň prahování, byla v [14] navržena parametrizovaná verze dynamického prahování pro algoritmus Immunos-99. V případě dynamického prahování nemá parametr t vliv. Proto byl tento parametr využit pro parametrizovanou verzi dynamického prahování. Původní způsob dynamického prahování dle (10) byl doplněn o parametr t následovně

$$Ab'_i = \{ \mathbf{ab}_j \mid \mathbf{ab}_j \in Ab_i; fit_j > |t| \times fit_{avg} \}. \quad (12)$$

Parametrem je tak možné změnit úroveň prahu pro každou třídu ve vztahu k dané průměrné hodnotě. Nastavení parametru t na zápornou hodnotu znamená použití parametrizované verze dynamického prahování dle (12).

4. Modelování bonity obcí

Analýza vlivu nastavení parametrů algoritmu Immunos-99 bude provedena na datovém souboru pro modelování bonity obcí. Bonita znamená schopnost a ochotu obce splácet svůj dluh. Použitý datový soubor obsahuje hodnocení bonity obcí Pardubického kraje [10], [11], [12], [13], [18] a je tvořen 12 popisnými proměnnými **Chyba! Nenalezen zdroj odkazů., Chyba! Nenalezen zdroj odkazů.,** které popisují 452 obcí Pardubického kraje. Hodnoty proměnných odpovídají stavu z roku 2004 a zahrnují dluhové, finanční a demografické ukazatele obcí. Navržený model je znázorněn na obr. **Chyba! Nenalezen zdroj odkazů.** V literatuře lze nalézt různé modely na bázi neuronových sítí, fuzzy množin nebo expertních systémů pro predikování a vysvětlení ratingu **Chyba! Nenalezen zdroj odkazů., Chyba! Nenalezen zdroj odkazů., Chyba! Nenalezen zdroj odkazů., Chyba! Nenalezen zdroj odkazů., Chyba! Nenalezen zdroj odkazů., Chyba! Nenalezen zdroj odkazů.**



Obr. 5: Modelování bonity obcí pomocí AIS

Datový soubor je rozdělen na trénovací a testovací množinu tak, že obě množiny mají stejné rozložení vzorů v jednotlivých třídách [15]. Trénovací data budou použita pro nastavení použitého algoritmu, který si pomocí těchto dat vytvoří vnitřní reprezentaci, kterou následně využije pro klasifikaci neznámých vzorů. Testovací data budou sloužit k ověření klasifikátoru a jeho interní reprezentaci dat. Vzory z trénovací množiny budou ohodnoceny klasifikátorem a výsledná klasifikace bude porovnána se správným přiřazením ke třídě.

Vzhledem k náhodné složce algoritmu Immunos-99 bude v rámci každého experimentu provedeno $n_{pok} = 10$ pokusů pro dané nastavení parametrů. Úspěšnost klasifikace i -tého pokusu Acc_i je dána tvarem

$$Acc_i = \frac{n_{cor,i}}{n_{test}}, \quad (13)$$

kde $n_{cor,i}$ je počet správně klasifikovaných vzorů z testovací množiny a n_{test} je počet vzorů v testovací množině. Každý experiment tak bude vyhodnocen průměrnou úspěšností klasifikace Acc_{avg} a maximální hodnotou úspěšnosti klasifikace Acc_{max} dosaženou na trénovací množině s daným nastavením. Experimenty byly realizovány v rámci prostředí WEKA [20] s využitím „Weka Classification Algorithms“ [6].

5. Návrh nastavení parametrů pro Immunos-99 pro modelování bonity obcí

Fáze učení algoritmu Immunos-99 je ovlivněna nastavením těchto parametrů [5], [14]:

- Parametr n_{init} [%] (velikost počáteční populace protilátek Ab_i) – udává velikost počáteční populace protilátek vztahenou k velikosti trénovací množiny.
- Parametr t (práh potlačení) – udává úroveň prahu potlačení protilátek ve fázi učení. Podle nastavené hodnoty lze rozlišit tyto typy prahování:
 - $t > 0$ – absolutní verze prahování
 - $t < 0$ – dynamické prahování s parametrem
 - $t = -1$ – dynamické prahování bez parametru. V podstatě se jedná o předchozí případ s fixovanou hodnotou parametru.
- Parametr n_{gen} (počet generací) – udává počet generací pro fázi učení.

Dále bude nejprve uveden návrh nastavení parametrů pro algoritmus Immunos-99. Vedle úspěšnosti klasifikace Acc bude sledovaným parametrem pro jednotlivá nastavení také výsledná velikost množiny paměťových protilátek vztahená relativně k velikosti trénovací množiny (množiny antigenů Ag). Sledovaný parametr tedy bude

$$r_m = \frac{|Ab_m|}{|Ag|} \times 100, \quad (14)$$

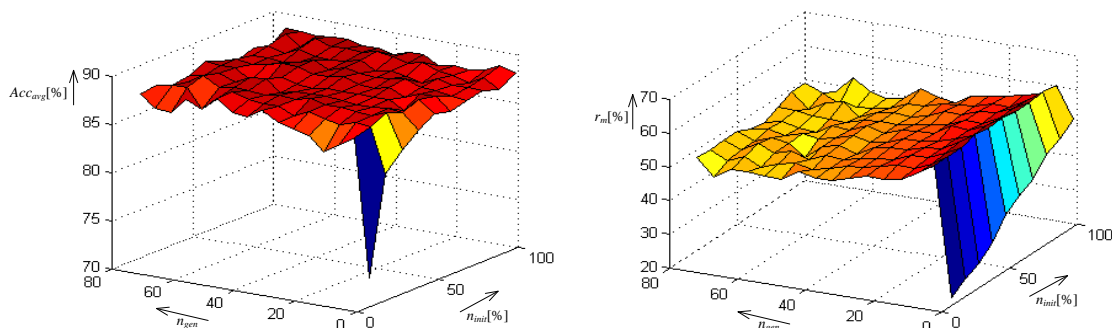
kde Ab_m je množina paměťových protilátek, která je výsledkem fáze učení a Ag je množina antigenů, která je vstupem fáze učení.

Klíčovým parametrem pro chování algoritmu Immunos-99 je úroveň potlačení protilátek – parametr t . V tab. 3 jsou uvedeny minimálních a maximálních hodnoty pro kvalitu jedinců pro úlohu modelování bonity obcí (předpoklad: velikost množiny protilátek a množiny antigenů je stejná). Jak je patrné, liší se rozsah hodnot pro jednotlivé třídy velmi výrazně. Největší rozdíl je patrný mezi rozsahem hodnot pro třídu 4 a třídu 7 (odpovídá největšímu rozdílu mezi počtem jedinců dané třídy v trénovací množině). Protože tento datový soubor má nerovnoměrně rozložené vzory mezi třídami, nelze použít absolutní verzi prahování. Pro tuto úlohu je možné použít pouze dynamickou verzi prahování bez parametru nebo s parametrem.

Tab. 3: Příklad rozsahu hodnot kvality protilátek pro datový soubor „Obce“

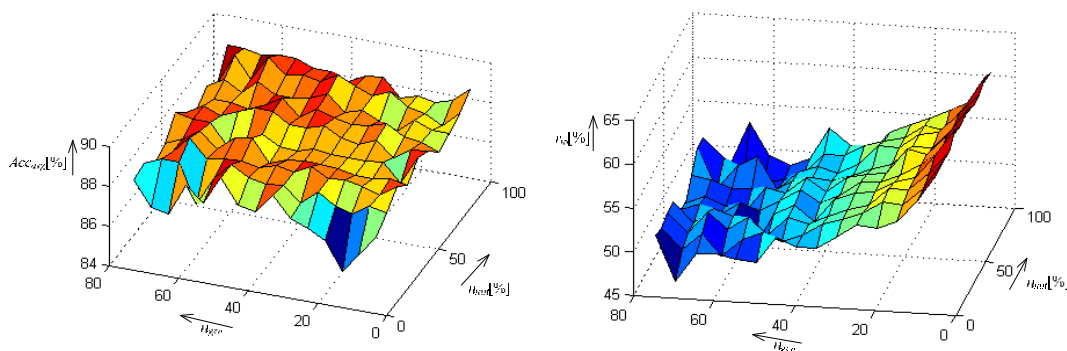
c_i	$ Ab_i $	$ Ag_i $	Kvalita protilátek dané třídy	
			$f_{min}(c_i)$	$f_{max}(c_i)$
1	24	24	0.0032	1.83
2	57	57	0.0035	11.52
3	97	97	0.0041	38.88
4	108	108	0.0043	50.49
5	31	31	0.0032	3.12
6	17	17	0.0031	0.90
7	5	5	0.0030	0.07

Na obr. 6 je uveden vliv nastavení počtu generací a počáteční velikosti populace při dynamickém prahování bez parametru. V případě nastavení parametru $n_{gen} = 1$ je patrný výrazný vliv nastavení parametru n_{init} , kdy algoritmus neměl možnost v rámci jednotlivých generací vytvořit správně populaci. Pokud zobrazíme průměrné dosažené výsledky od počtu generací $n_{gen} = 10$ (viz obr. 7), je patrný pokles počtu paměťových protilátek ve výsledné populaci s rostoucím počtem generací. Jak je patrné, tak nejlepších průměrných výsledků klasifikace bylo dosaženo s nejmenším počtem protilátek ve výsledné množině. Algoritmus v rámci fáze učení optimalizoval počet protilátek tak, že kvalitních výsledků dosáhl s menším počtem protilátek v rámci vnitřní reprezentace. Vlivem jemnějšího měřítka je patrný rozptyl průměrné úspěšnosti klasifikace v závislosti na nastavení parametrů n_{gen} a n_{init} , který se pohybuje v rozmezí cca $\pm 1\%$.



Obr. 6: Výsledky algoritmu Immunos-99 na základě parametrů n_{init} a n_{gen} při dynamickém prahování

a) průměrná úspěšnost klasifikace, b) velikost výsledné populace protilátek r_m



Obr. 7: Výsledky algoritmu Immunos-99 na základě parametrů n_{init} a n_{gen} (od $n_{gen} = 10$) při dynamickém prahování a) průměrná úspěšnost klasifikace, b) velikost výsledné populace protilátek r_m

Z průběhů na obr. 6 a obr. 7 lze říci, že od počtu 20 generací se dosažené výsledky ustálí a zvyšováním počtu generací již nedochází k výraznému zvýšení kvality dosažených výsledků. Nadále však dochází k poklesu velikosti výsledné populace. Velikost výsledné populace se ustálí až mezi 40 a 50 generacemi. S přihlédnutím k oběma uvedeným skutečnostem lze v tomto případě doporučit nastavení parametru $n_{gen} = \langle 30; 50 \rangle$. Pro nastavení parametru n_{init} lze vzhledem k chování algoritmu při malém počtu generací doporučit vysoké hodnoty $n_{init} = \langle 70\%; 100\% \rangle$. V případě použití algoritmu Immunos-99 s dynamickým prahováním bez parametru pro modelování bonity obcí lze doporučit následující nastavení parametrů:

- Parametr t (práh) – V případě dynamického prahování bez parametru je parametr t konstantní $t = -1$.
- Parametr n_{init} (počáteční velikost populace) – Doporučené počáteční nastavení parametru tohoto parametru je v rozsahu $n_{init} = \langle 70\%; 100\% \rangle$ z velikosti trénovací množiny.
- Parametr n_{gen} (počet generací) – Na základě provedených experimentů lze pro modelování bonity obcí doporučit nastavení parametru $n_{gen} = \langle 30; 50 \rangle$ s následnou optimalizací nastavení tohoto parametru s ohledem na dosažené výsledky a velikost výsledné populace.

V tab. 5 jsou uvedeny tři nejlepší průměrné výsledky úspěšnosti klasifikace získané v rámci provedených experimentů pro dynamické prahování bez parametru. Tři nejlepší výsledky podle maximální úspěšnosti klasifikace jsou uvedeny v tab. 5. Z uvedených výsledků lze říci, že nejlepší výsledky (ve smyslu průměrných a maximálních hodnot úspěšnosti klasifikace) dosahuje algoritmus Immunos-99 při vysokých hodnotách velikosti počáteční populace a vyšším počtu generací.

Tab. 4: Tři nejlepší dosažené výsledky podle průměrné úspěšnosti klasifikace pro dynamické prahování

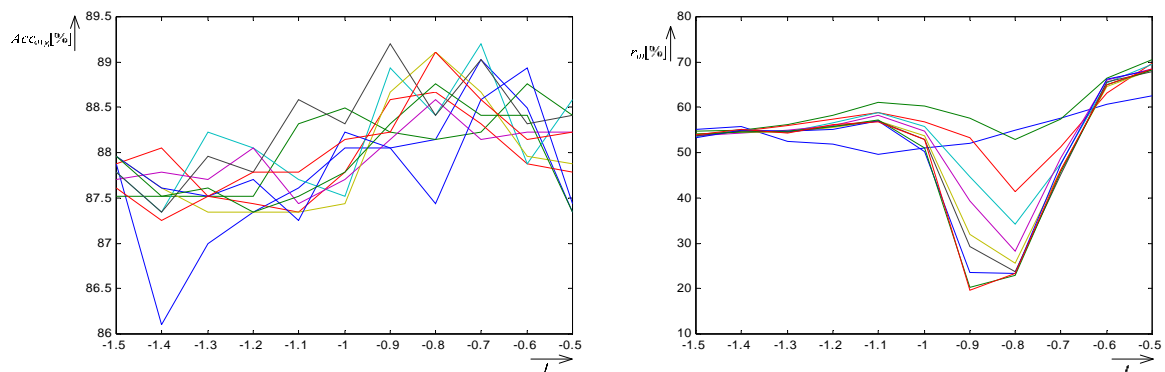
$Acc_{min}[\%]$	$Acc_{avg}[\%]$	$Acc_{max}[\%]$	□	$r_m[\%]$	n_{gen}	$n_{init}[\%]$
86.73	89.76	90.27	1.06	51.06	40	90
87.61	88.50	89.38	0.56	50.29	40	80
87.61	88.50	89.38	0.69	60.29	5	100

Tab. 5: Tři nejlepší dosažené výsledky podle maximální úspěšnosti klasifikace pro dynamické prahování

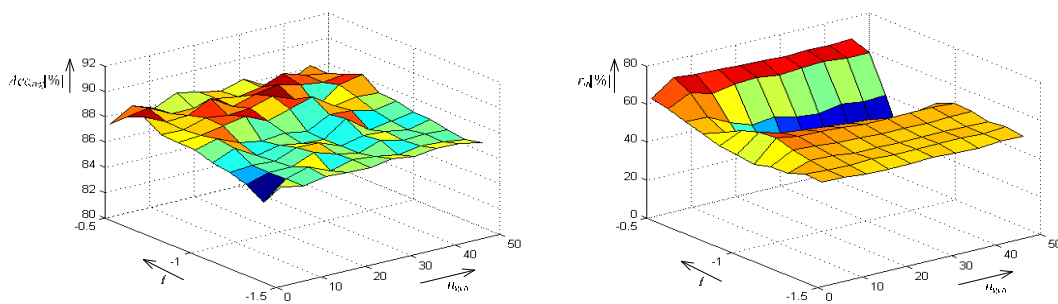
$Acc_{min}[\%]$	$Acc_{avg}[\%]$	$Acc_{max}[\%]$	\square	$r_m[\%]$	n_{gen}	$n_{init}[\%]$
86.73	89.76	90.27	1.06	51.06	40	90
86.73	88.32	90.27	1.11	59.79	5	30
85.84	88.23	90.27	1.19	51.00	45	90

Dále je uveden návrh nastavení parametrů pro algoritmus Immunos-99 s dynamickým prahováním s parametrem. Na obr. 8 a obr. 9 je znázorněn vliv nastavení parametru dynamického prahování na průměrnou úspěšnost klasifikace a velikost výsledné populace pro různé hodnoty počtu generací v rozsahu $n_{gen} = \langle 1; 50 \rangle$, při konstantní hodnotě $n_{init} = 100\%$.

Jak je z uvedených obrázků patrné, tak nejlepších hodnot průměrné úspěšnosti klasifikace je dosaženo pro nastavení parametru v rozsahu $\langle -0.9; -0.7 \rangle$. I v tomto případě je vidět, že úspěšnost klasifikace má nepřímou závislost na velikosti výsledné populace paměťových protilátek. Nejlepších výsledků je zde dosaženo s nízkým počtem protilátek ve výsledné populaci ve vztahu k velikosti trénovací množiny. Algoritmus v rámci fáze učení vygeneroval populaci s menším počtem protilátek, která ale velmi kvalitně reprezentuje trénovací množinu i při použití testovacích dat.



Obr. 8: Parametrizované dynamické prahování – úspěšnost klasifikace v závislosti na nastavení prahu t pro $n_{gen} = \langle 1; 50 \rangle$



Obr. 9: Parametrizované dynamické prahování – úspěšnost klasifikace v závislosti na nastavení prahu t a počtu generací n_{gen}

V tab. 6 jsou uvedeny tři nejlepší dosažené průměrné výsledky úspěšnosti klasifikace získané pomocí parametrizované verze dynamického prahování algoritmu Immunos-99 v rámci provedených experimentů. Nejlepších hodnot průměrné úspěšnosti klasifikace bylo

vždy dosaženo s doporučenou hodnotou parametru t a vysokou hodnotou velikosti počáteční populace n_{init} . Počet generací ve srovnání s verzí bez parametru klesl.

Tab. 6: Nejlepší průměrná úspěšnost klasifikace – dynamický práh s parametrem

$Acc_{min}[\%]$	$Acc_{avg}[\%]$	$Acc_{max}[\%]$	\square	t	n_{gen}	$n_{init}[\%]$
87.61	90.00	91.15	1.12	-0.9	25	90
87.61	89.47	91.15	1.15	-0.7	25	80
87.61	89.38	91.15	1.25	-0.7	6	80

V tab. 7 jsou uvedeny tři nejlepší maximální výsledky klasifikace Acc_{max} , které byly získány s doporučeným nastavením prahu t pro algoritmus Immunos-99 s dynamickým prahováním s parametrem. Stejně jako v tab. 6. klesl celkový počet generací. V jednom z uvedených případů klesla i počáteční velikost populace protilátek.

Tab. 7: Nejlepší maximální úspěšnost klasifikace – dynamický práh s parametrem

$Acc_{min}[\%]$	$Acc_{avg}[\%]$	$Acc_{max}[\%]$	\square	t	n_{gen}	$n_{init}[\%]$
86.73	89.38	92.04	1.43	-0.9	15	80
84.96	89.29	92.04	2.22	-0.8	25	70
86.73	89.20	92.04	1.89	-0.9	30	50

V tab. 8 je uvedeno srovnání nejlepších výsledků z experimentů s dynamickým prahováním bez parametru a výsledků z experimentů s dynamickým prahováním s parametrem. Jak je patrné, tak jak v případě průměrných, tak i maximálních hodnot došlo ke zvýšení úspěšnosti klasifikace. V rámci provedených experimentů tak navržená varianta dynamického prahování s parametrem dosahuje lepších výsledků, než originální dynamické prahování bez parametru.

Tab. 8: Srovnání výsledků pro dynamické prahování bez a s parametrem

	Dynamické prahování bez parametru	Dynamické prahování s parametrem
Nejlepší $Acc_{avg} [\%]$	89.76	90.00
Nejlepší $Acc_{max} [\%]$	90.27	92.04

Z hodnot uvedených v tab. 6 a v tab. 7 je patrný pokles potřebného počtu generací. Na základě provedených experimentů lze tedy v případě dynamického prahování s parametrem snížit doporučený rozsah počtu generací pro úlohu modelování bonity obcí. Doporučené nastavení počáteční velikosti populace zůstane zachováno. V případě použití algoritmu Immunos-99 s dynamickým prahováním s parametrem pro modelování bonity obcí lze doporučit následující nastavení parametrů:

- Parametr t (práh) – V případě dynamického prahování s parametrem je doporučené nastavení v rozsahu $\langle -0.9; -0.7 \rangle$.
- Parametr n_{init} (počáteční velikost populace) – Doporučené počáteční nastavení parametru tohoto parametru je v rozsahu $n_{init} = \langle 70\%; 100\% \rangle$ z velikosti trénovací množiny.
- Parametr n_{gen} (počet generací) – Na základě provedených experimentů pro dynamické prahování s parametrem lze pro tuto úlohu doporučit nastavení parametru

$n_{gen} = \langle 15; 30 \rangle$ s následnou optimalizací nastavení tohoto parametru s ohledem na dosažené výsledky a velikost výsledné populace.

6. Závěr

Tento článek se zaměřil na algoritmus Immunos-99, konkrétně na chování tohoto algoritmu ve vazbě na modelování bonity obcí. Nejprve byly uvedeny základní vlastnosti tohoto algoritmu. V rámci fáze učení je hodnocena kvalita protilátky \mathbf{ab}_j ve vazbě na antigen \mathbf{ag} . Tato kvalita ale není funkcí pouze této protilátky a antigenu, ale také velikosti a rozložení množiny protilátek Ab_i , do které \mathbf{ab}_j náleží a rozložení a velikosti množiny antigenů Ag . Tato závislost se pak negativně projevuje v rámci tzv. absolutního prahování, kdy může dojít k plnému potlačení protilátek jedné třídy. V originální variantě algoritmu Immunos-99 [5] je tento vliv omezen pomocí dynamického prahování (bez parametru). Tím dojde ale ke ztrátě možnosti ovlivnit úroveň potlačení protilátek. Aby při dynamickém prahování (bez parametru) nedošlo ke ztrátě možnosti ovlivnit úroveň prahování, byla navržena parametrizovaná verze dynamického prahování pro algoritmus Immunos-99. Tato varianta prahování využívá obou výhod, umožňuje ovlivnit úroveň potlačení a zároveň při nerovnoměrném rozložení vzorů v trénovací množině nedochází k úplnému potlačení protilátek jedné třídy.

Pro úlohu modelování bonity obcí bylo dále navrženo nastavení parametrů. Nejprve byl uveden návrh nastavení parametrů pro variantu dynamického prahování bez parametru. Následně bylo navrženo nastavení pro dynamické prahování s parametrem. Na základě provedených experimentů pro úlohu modelování bonity obcí lze říci, že navržený způsob dynamického prahování s parametrem dosahuje lepších hodnot úspěšnosti klasifikace, jak z hlediska maximálních, tak i průměrných hodnot úspěšnosti klasifikace. Dále při dynamickém prahování s parametrem je zapotřebí menší počet generací pro fázi učení, což snižuje výpočetní náročnost tohoto algoritmu.

Na základě získaných výsledků lze konstatovat, že pro úlohu modelování bonity obcí lze algoritmus Immunos-99 použít a to zejména navrženou variantu dynamického prahování s parametrem.

Použitá literatura:

- [1] BRABAZON, A., O'NEILL, M. Credit Classification Using Grammatical Evolution. *Informatica*, 2006, no. 30, pp. 325-335.
- [2] BRENNAN, D., BRABAZON, A. Corporate Bond Rating Using Neural Networks. In *Proceedings of the International Conference on Artificial Intelligence, IC-AI '04, June 21-24, 2004, Las Vegas, Nevada, USA*. CSREA Press, 2004, vol. 1, pp. 161-167.
- [3] BROWLEE, J. *Clonal Selection Algorithms*. Technical Report, no. 070209A, Center for Intelligent Systems and Complex Processes (CISCP), Faculty of Information and Communication Technologies, Swinburne University of Technology, Australia. February 2007.
- [4] BROWLEE, J. *Clonal Selection Theory & Clonalg - The Clonal Selection Classification Algorithm (CSCA)*. Technical Report, no. 2-02, Center for Intelligent Systems and Complex Processes (CISCP), Faculty of Information and Communication Technologies, Swinburne University of Technology, Australia. January 2005.
- [5] BROWLEE, J. *Immunos-81, The Misunderstood Artificial Immune System*. Technical Report, no. 3-01, Center for Intelligent Systems and Complex Processes (CISCP), Faculty of Information and Communication Technologies, Swinburne University of Technology, Australia. January 2005.

- [6] BROWLEE, J. *WEKA Classification Algorithms* [počítačový program]. Ver. 1.6. Sourceforge.NET, 2006 [citováno 2008-09-09]. Dostupné z <<http://sourceforge.net/projects/wekaclassalgos>>.
- [7] CARTER, J. H. The Immune System as a Model for Classification and Pattern Recognition. *Journal of the American Medical Informatics Association*, 2000, vol. 7, no. 1, pp. 28-41.
- [8] CASTRO, L. N. - TIMMIS, J. I. *Artificial Immune Systems: A New Computational Intelligence Approach*, London: Springer-Verlag, 2002. 357 p. ISBN 1-85233-594-7.
- [9] CASTRO, L. N., ZUBEN, F. J. Learning and Optimization Using the Clonal Selection Principle. *IEEE Transactions on Evolutionary Computation, Special Issue on Artificial Immune Systems*. 2002, vol. 6, no. 3, pp. 239-251. ISSN 1089-778X.
- [10] HÁJEK, P., OLEJ, V. Hierarchical Structure of Fuzzy Inference Systems Design for Municipal Creditworthiness Modeling. *WSEAS Transaction on Systems and Control*, February 2007, vol. 2, no. 2, pp.162-169. ISSN 1991-8763.
- [11] HÁJEK, P., OLEJ, V. Modelling Municipal Rating by Cluster Analysis and Neural Networks. In *Proceedings of the 7-th WSEAS International Conference on Neural Networks, NN 2006, Cavtat, Croatia, June 12-14*. Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS), 2006, pp.73-78. ISBN 960-8457-46-7.
- [12] HÁJEK, P., OLEJ, V. Municipal Creditworthiness Modelling by Kohonen's Self-Organizing Feature Maps and LVQ Neural Networks. In *Proceedings of the 9th Eighth International Conference on Artificial Intelligence and Soft Computing, ICAISC 08, Lecture Notes in Artificial Intelligence, Zakopane, Poland, June 22-26*, Springer Berlin Heidelberg New York, 2008, pp. 52-61. ISSN 0302-9743, ISBN 3-540-69572-9.
- [13] HÁJEK, P., OLEJ, V. Municipal Creditworthiness Modelling by Clustering Methods. In *Proceedings of the 10th International Conference on Engineering Applications of Neural Networks, EANN 2007, Margaritis, Iliadis, Eds., Thessaloniki, Greece, August 29-31*. 2007, pp.168-177, ISBN 978-960-287-093-8.
- [14] KOPÁČEK, L. – OLEJ, V. Návrh algoritmů na bázi umělých imunitních systémů pro úlohy klasifikace. *Scientific Papers of the University of Pardubice, Faculty of Economics and Administration 13*, Series D. Pardubice: Univerzita Pardubice, 2008. s. 64-78. ISBN 978-80-7395-149-8.
- [15] KVASNIČKA, V. a kol. *Úvod do teorie Neuronových sítí*. Bratislava: IRIS, 1997. ISBN 80-88778-30-1.
- [16] LIU, X., LIU, W. Credit Rating Analysis with AFS Fuzzy Logic. *Advances in Natural Computation, Lecture Notes in Computer Science, Springer-Verlag*, 2005, vol. 3612, pp. 1198-1204. ISSN 0302-9743.
- [17] MAGNI, C.A. Rating and Ranking Firms with Fuzzy Expert Systems: The Case of Camuzzi. *MPRA Paper University Library of Munich, Germany*, 2007, vol. 5889.
- [18] OLEJ, V., HÁJEK, P. Modeling Municipal Rating by Unsupervised Methods. *WSEAS Transactions on Systems*, July 2006, vol. 5, no. 7, pp.1679-1686, ISSN 1109-2777.
- [19] ROMANIUK, S.G. *Fuzzy Rule Extraction for Determining Creditworthiness of Credit Applicants*. Technical report, No. TR20/92, National University of Singapore, 1992.
- [20] WITTEN, I. H., FRANK, E. *Data Mining: Practical Machine Learning Tools and Techniques*. Second Edition. Morgan Kaufmann, 2005. ISBN 0-12-088407-0.
- [21] ZHOU, X.Y., ZHANG, D.F., JIANG, Y. A New Credit Scoring Method Based on Rough Sets and Decision Tree. *Advances in Knowledge Discovery and Data Mining, Lecture Notes in Computer Science, Springer-Verlag*, 2008, vol. 5012, pp. 1081-1089. ISSN 0302-9743.

Kontaktní adresa:

Ing. Luděk Kopáček
Univerzita Pardubice
USII/FES
Studentská 84, Pardubice, 53210
e-mail: ludek.kopacek@centrum.cz